

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA



Robotic Process Automation – Desenvolvimento de Soluções Automatizadas

Inês Santos Ferreira

Mestrado em Informática

Trabalho de Projeto orientado por:
Carlos Eduardo Ramos dos Santos Lourenço

Resumo

A automação robótica de processos combina a utilização de inteligência artificial com a automação por forma a ganhar agilidade, produtividade, fiabilidade e redução de custo operacionais nos fluxos de trabalho. É uma ferramenta que permite auxiliar as empresas a concretizar a sua transformação digital, sendo esta transformação algo imprescindível para a manutenção da sua competitividade especialmente atendendo à pandemia, que forçou uma remodelação do uso da tecnologia no contexto laboral. [20]

A automatização de processos está a ser adotada em várias indústrias entre as quais se inclui a banca e as telecomunicações.

O seu uso é especialmente realçado na eficiência em casos de tarefas repetitivas e monótonas que eram anteriormente realizadas por uma *workforce* humana, permitindo às equipas das áreas que utilizam esta tecnologia focarem-se em tarefas nas quais é realmente necessária uma análise humana. [19]

Este relatório centra-se precisamente em dois projetos de automação de processos: o DuiT, que tem como objetivo a implementação e desenvolvimento de automatismos no âmbito de uma entidade financeira e o UAI, cujos objetivos são os mesmos com a diferença de serem aplicados a uma empresa de telecomunicações. Deste modo, o presente relatório apresenta a abordagem utilizada para identificar, selecionar e implementar diversas oportunidades de automatização com uso da ferramenta Blue Prism. A fase de desenvolvimento e implementação tomam especial foco dado ao facto de serem as atividades que me foram assignadas enquanto membro da equipa DuiT e da equipa UAI no âmbito do estágio.

Assim, todas as fases de desenvolvimento e respetivos entregáveis serão alvo de maior detalhe no presente relatório.

Ambos os projetos funcionam com base na metodologia Agile, pelo que são completados vários *deliverables* ao longo do decorrer do estágio.

Palavras-chave: RPA, Automatização de processos, Inteligência Artificial, Blue Prism, Força de Trabalho Digital

Abstract

Robotic process automation combines the use of artificial intelligence with automation as a way to earn agility, productivity, reliability and operational cost reduction on the work fluxes. It is a tool that allows companies to transform digitally, which is fundamental for keeping their competitiveness especially given the pandemic, which has forced a change of the use of technology in the working context. [20]

Process automation is being adopted into several industries from which banking and telecommunications are amongst.

Its use is highlighted on its efficiency on cases in which repetitive and monotonous tasks were previously performed by a human workforce, allowing the areas' teams that use this technology to focus on tasks that require human analysis. [19]

This report's focus relies precisely on two process automation projects: DuiT, which has the objective of implementing and developing automations with its scope being in a financial institution, and UAI, which shares the same objectives with the difference that they regard a telecommunication provider.

This way, the report presents the approach used to identify, select, and implement several automation opportunities with the use of the Blue Prism tool. The development and implementation phases are the main focus as they are the activities that I was assigned as a member of the DuiT and UAI teams in the training scope.

Because of this, all the developments and correspondent deliverables will be specially focused upon on the report.

Both projects work with the Agile methodology, so several deliverables are completed as the training occurs.

Keywords: RPA, Process Automation, Artificial Intelligence, Blue Prism, Digital Workforce

Conteúdo

Capítulo 1	Introdução	10
1.1	Motivação	10
1.2	Objetivos.....	10
1.3	Organização do documento	11
Capítulo 2	Trabalho relacionado.....	12
Capítulo 3	O trabalho – Metodologia e Ciclo de Entrega	15
3.1	Metodologia	15
3.2	A Ferramenta	18
3.3	Identificação de oportunidades	21
3.4	Seleção da Oportunidade	23
3.5	Levantamento de Requisitos	23
3.6	Documentação.....	24
3.6.1	Desenho Funcional	24
3.6.2	Process Development Design (PDD).....	25
3.7	Desenvolvimento	26
3.8	Testes.....	27
3.9	Métricas	28
3.10	Monitorização e Estabilização	28
Capítulo 4	Trabalho Realizado	30
4.1	Projeto DuiT	30
4.1.1	Oportunidade P.15 – Impressão de Certificados de Formação (Sprint 19.8)	31
4.1.2	Oportunidade P.60 – Despesas Reclamadas a Bancos (Sprint 19.9).	40
4.2	Projeto UAI.....	45
4.2.1	Oportunidade 0002 – LAC BO	46
4.2.2	Discovery e Shadowings.....	57
4.2.3	Automatismo KPI_ReportBuilder	57
4.2.4	Code Review	59

4.3	Planeamento/Calendarização	60
Capítulo 5	Discussão	63
	Bibliografia.....	70
	Apêndice	73
A.	Glossário	73
B.	Acrónimos e Siglas	75

Índice de Figuras

Figura 1. Agile no contexto da automatização.	15
Figura 2. Gestão de Delivery.....	17
Figura 3. Action stage.....	19
Figura 4. Inicialização de uma variável com uso de Calculation Stage.....	18
Figura 5. Code Stage.....	19
Figura 6. Fila de trabalho com casos de sucesso e exceção.	20
Figura 7. Avaliação da complexidade de uma oportunidade [1]. Nota: Citrix corresponde à necessidade de utilização de uma ferramenta de virtualização de aplicações ou ‘desktops’ para permitir a interação com a aplicação do cliente.	22
Figura 8. Exemplo de um fluxo de atividades.....	24
Figura 9. Exemplo de um template constante de um documento funcional.	25
Figura 10. Exemplo de um Dashboard contendo métricas.	25
Figura 11. Exemplo da estruturação de tarefas na aplicação Trello.....	26
Figura 12. Planeamento inicial dos primeiros três sprints do estágio.....	31
Figura 13. Milestones da oportunidade P.15.....	32
Figura 14. Fluxo de atividades da oportunidade P.15.....	33
Figura 15. Criação de uma ação de formação na plataforma SIGO.	35
Figura 16. Aplicação CutePDF Pro para a inserção de carimbo.	37
Figura 17. Dashboard contendo as métricas a oportunidade P.15 (povoado com dados de teste das execuções de qualidade).	39
Figura 18. Fluxo de atividades da oportunidade P.60.....	41
Figura 19. Milestones da oportunidade P.60.....	41
Figura 20. Dashboard contendo as métricas a oportunidade P.60 (dados teste de execuções em qualidade).....	44
Figura 21. Fluxo de atividades	46
Figura 22. Página Submain estruturada com máquina de estados.....	50
Figura 23. Nº de solicitações (casos) processados por dia.....	56
Figura 24. Fluxo de atividades do mecanismo KPI_ReportBuilder.	58
Figura 25. Planeamento Inicial.....	60
Figura 26. Mapa temporal da oportunidade 0002 - LAC BO. A azul claro está a previsão inicial e a azul escuro o tempo real consumido.	61

Índice de Tabelas

Tabela 1. Glossário	74
Tabela 2. Acrônimos e Siglas	75

Capítulo 1

Introdução

1.1 Motivação

Dada a realidade tecnológica atual, as empresas e sectores devem adaptar-se cada vez mais rapidamente à constante evolução e demanda da parte dos seus clientes. Por este motivo, a transformação digital deixa de ser apenas um ideal, mas sim uma necessidade.

Dois dos grandes sectores que tem de momento bastante potencial de digitalização são a banca e as telecomunicações, e um dos métodos para alcançar esse objetivo é a automatização de processos.

A automatização permite uma melhoria na rapidez da execução de um processo, a sua fiabilidade, redução de custos operacionais, entre outros, para além de permitir libertar recursos para a execução de outras tarefas.

Em ambos os sectores, é frequente encontrar tarefas monótonas, repetitivas e com um conjunto de regras bem definidas que não concretizam o potencial dos colaboradores que as executam, pelo que este projeto, realizado numa instituição financeira e numa empresa de telecomunicações clientes da Accenture, irá permitir a organização de processos, para além de outros benefícios já referidos e uma distribuição mais eficiente das tarefas dos seus colaboradores.

1.2 Objetivos

O objetivo do corrente projeto passa por executar estratégia de *Robotic Process Automation* (RPA), configurando os processos, padrões e modelos de execução de acordo com a estratégia alinhada com o cliente.

Mais especificamente, o objetivo do estágio passa por integrar a equipa de *developers* de RPA's da Accenture na gestão de *delivery* e respetivas fases para desenvolver e entregar vários módulos de soluções automatizadas, *templates*, cadernos de testes, desenhos funcionais e *dashboards* respeitantes às oportunidades selecionadas.

1.3 Organização do documento

Este documento está dividido em quatro capítulos. O primeiro enquadra a informação introdutória ao documento, contendo a motivação e objetivos deste. Seguidamente, o segundo capítulo inclui uma breve descrição do trabalho realizado na área da automatização de processos. O terceiro descreve a metodologia e gestão de *delivery* (entrega). O grosso do relatório encontra-se centrado no quarto capítulo, que contém a informação acerca do trabalho realizado no âmbito do estágio. Contém também uma breve descrição da ferramenta utilizada e a metodologia em uso. Este capítulo inclui ainda as várias fases contempladas na automatização de uma oportunidade e posteriormente a descrição das atividades realizadas nos projetos e automatizações associadas.

O capítulo cinco engloba a discussão dos resultados obtidos no que diz respeito às métricas, análise do cumprimento das datas propostas e questões de segurança.

No apêndice é possível encontrar o glossário e uma lista de siglas e acrónimos.

Capítulo 2

Trabalho relacionado

Em termos do mercado de automação, três das empresas que atualmente dominam o mercado tratam-se da Automation Anywhere, UiPath e a Blue Prism, que conta como parceiros empresas como Accenture, Google, Microsoft, IBM, entre outros. [17]

Entre as três ferramentas Automation Anywhere, UiPath e a Blue Prism, verificam-se algumas diferenças. Estas incluem desde a arquitetura, em que a Automation Anywhere e Blue Prism usam um servidor enquanto a UiPath utiliza um orquestrador *web* ou em *cloud*; o preço, em que o Blue Prism se torna mais acessível entre as três; a facilidade de aprendizagem e o nível de experiência que o *developer* necessita, em que se verifica que para a Automation Anywhere é necessário um nível intermédio a alto, para a Blue Prism e UiPath, é necessário um nível básico e a forma de apresentação/desenvolvimento, em que tanto o UiPath como o Blue Prism apresentam o código sobre a forma de um fluxograma. Estas são diferenças a ter em conta quando se pretende seleccionar a ferramenta de automatização. [16]

Estas ferramentas permitem automatizar processos de negócio sem que seja necessário alterar as aplicações ou software em utilização pelo cliente, que nem sempre está preparado para atualizar alguns sistemas centrais à sua atividade que muitas vezes são múltiplos e não partilham informação entre estes. Por outro lado, uma vez que um processo pode vir a interagir com várias aplicações, o tempo de atualização do software em uso muitas vezes é bastante superior à implementação de RPAs. Existe também a vantagem de que, assumindo que o seu desenvolvimento ocorreu de forma modular, é possível adaptar a secção do processo automatizado que sofreu modificações às novas versões.

No entanto, existem também desvantagens em relação à informatização de processos/atualizações de aplicações. O automatismo (também referido como agente digital ou Robot) funciona tendo como base a replicação das ações de um operador, ou seja, faz uso das aplicações sem as alterar (ex: leitura, validação, inserção e submissão de dados), não substituindo as interfaces ou conteúdo das aplicações em uso. Assim, não impede a existência de dívida tecnológica caso seja implementado para interagir com uma aplicação que careça de atualização/otimização. Esta situação leva a uma segunda desvantagem, uma vez que se a aplicação sofrer alterações, pode ser

necessário reconfigurar a parte do automatismo que interage com a mesma (ex: se o automatismo esperar um botão cujo valor seja ‘Submeter’ e este passar a ter o valor ‘Inserir Dados’, este botão deixa de ser reconhecido e o automatismo falha), o que implica a necessidade de existir um modelo de manutenção evolutiva.

Por outro lado, para além do elevado investimento inicial, torna-se também necessário garantir a cooperação de operadores humanos, que frequentemente não vêem as automatizações como uma ferramenta que traz qualidade ao seu trabalho eliminando a necessidade da realização de tarefas de carácter repetitivo e com pouca necessidade de decisão humana. Existe naturalmente uma desconfiança em relação a esta tecnologia assim como outras que sejam vistas como ameaça ao seu posto de trabalho. Normalmente, mais do que uma redução de trabalhadores, estes são realocados para outras tarefas de valor acrescentado que necessitam de análise humana, mas nem sempre é o caso.

A ferramenta Blue Prism, por ser a que será utilizada, será discutida posteriormente neste trabalho.

Entre casos de sucesso na utilização de automação, encontra-se a Bancolombia, na qual existem automatismos de gestão de portfólios de investimento, a Zurich Insurance, com um exemplo de atribuição de políticas, a Keybank com automatismos de geração de recibos, e a Guardian Group, que fez uso desta tecnologia para reduzir os seus sistemas *legacy*. [21]

Anteriormente à data de início deste estágio, a Accenture também vem a trabalhar com diversas instituições financeiras as quais empregam o uso da automação, especificamente através do uso da ferramenta Blue Prism a qual incorpora no seu site casos de estudo em empresas como a Coca-Cola, Siemens, Ageas, Walgreens, Shop Direct, entre outros. [1]

No caso da Coca-Cola, os serviços de Recursos Humanos e Finanças utilizaram os recursos virtuais para passar de turnos de 8 para 24 horas sem alterar o número de trabalhadores, aumentando a produtividade. A utilização da ferramenta Blue Prism permitiu que os trabalhadores humanos pudessem focar-se em tarefas para resolução de situações que podem afetar os clientes e focarem-se em operações de negócio. [13]

Para a seleção dos processos a otimizar, a Coca-Cola começou pela área de Recursos Humanos, recolhendo dados como volume dos processos, frequência e número de vezes que estes são transferidos entre pessoas e/ou áreas. A recolha de dados permitiu assim atribuir uma pontuação aos processos descobertos de forma a detetar quais tinham maior potencial ao avaliar viabilidade para automatizar, qual o nível de risco e qual o número de pessoas envolvidas. [13]

Após a construção das soluções, foi possível alterar o âmbito das tarefas dos trabalhadores, que puderam assim passar a focar-se em tarefas que necessitam de análise em vez de tarefas como o processamento manual de dados. [13]

Outro caso onde a ferramenta Blue Prism foi empregue trata-se da empresa Shop Direct. A empresa pretendia tornar as suas tarefas mais eficientes e efetivas. Alguns dos desafios incluíam melhorar a agilidade do negócio em resposta a alterações, gestão do impacto de picos e quedas sazonais e simplificação dos processos administrativos.

A Blue Prism providenciou uma solução escalável e auditável de automação apoiada por serviços que permitiram uma forma rápida e adaptável de automatizar processos, conseguindo auxiliar a Shop Direct a criar um ambiente de operação *agile*, a redução de FTEs (*Full Time Equivalents*, 1 FTE corresponde à carga de trabalho de um recurso humano) necessários para administrar processos manuais, libertando recursos para a realização de tarefas de valor acrescentado e melhorando a qualidade do processamento. Uma das vantagens da utilização desta ferramenta foi não necessitar de alterações em sistemas previamente existentes e o alívio sobre os recursos de IT. [15]

Capítulo 3

O trabalho – Metodologia e Ciclo de Entrega

3.1 Metodologia

Neste projeto a metodologia em uso é a Agile, na qual se aposta em iterações (*sprints*) curtos (intervalos de entre 1 a 4 semanas ao final das quais se pretende a entrega de um *minimum viable product*, MVP, que corresponde normalmente a um entregável ou requisito) na interação com os *stakeholders* para garantir a qualidade do produto final.

A metodologia Agile permite obter *feedback* ao longo das várias fases de implementação da solução, aproximando tanto quanto possível o desenvolvimento aos requisitos pedidos. Permite também uma resposta mais rápida a pedidos de alteração ou adaptação a situações imprevistas ou não identificadas a fase durante o levantamento de requisitos (ver Figura 1).

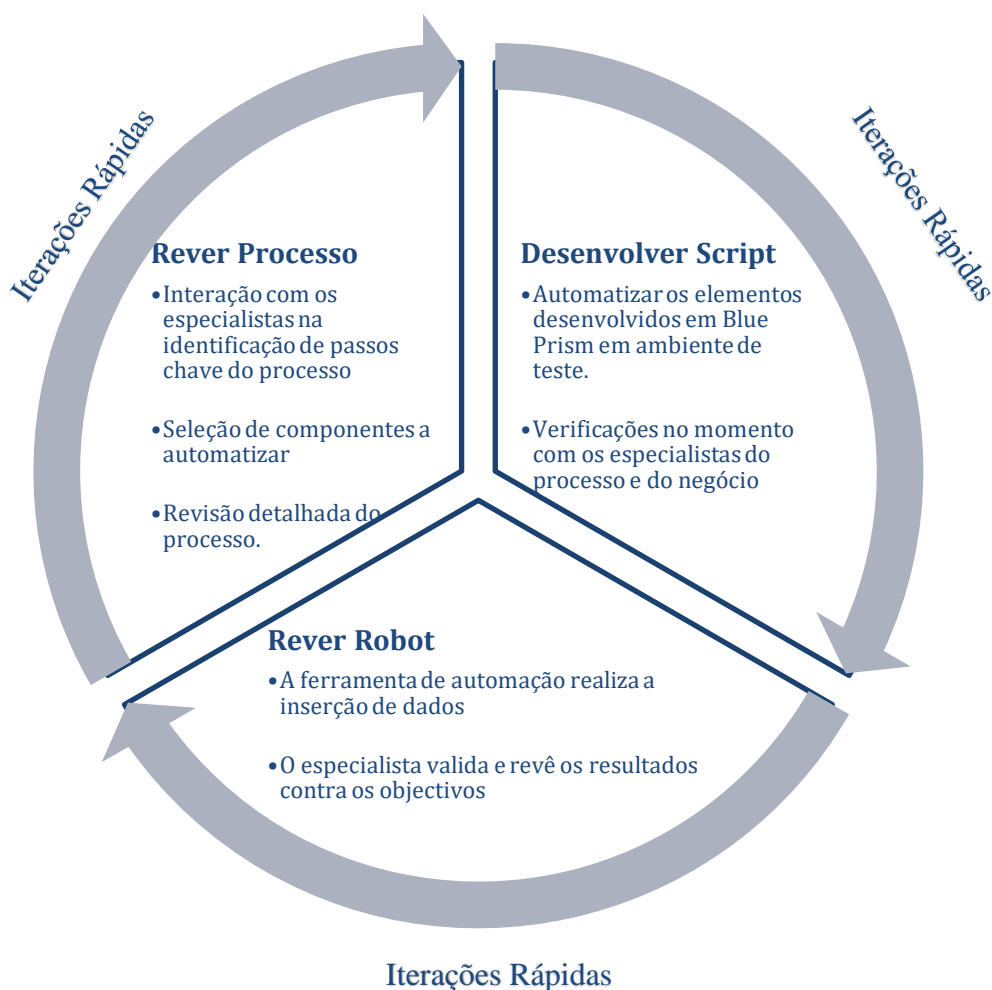


Figura 1. Agile no contexto da automação.

Para realizar a entrega de automatismos, existem várias fases necessárias para produzir os entregáveis necessários estas estão compreendidas na gestão de entrega (*delivery*) (ver Figura 2).

A gestão de entrega das soluções automatizadas é composta pelas seguintes fases:

1. Descoberta – Na qual são avaliadas as oportunidades de automação e respetivos *business cases*, *stakeholders*, requisitos e onde é realizada uma planificação *high level* das automações por forma a validar a sua viabilidade. Alguns dos dados analisados são os tempos médios de operação, volumetria, FTEs, complexidade do desenvolvimento, entre outros.
2. Design – Na qual são realizadas reuniões de levantamento de requisitos detalhado do qual resulta o desenho funcional. Posteriormente, é contruído o desenho técnico da solução assim como o respetivo plano de testes e são analisados os cenários possíveis do decorrer do processo. Finalmente, ocorre a fase de validação do desenho. São também aqui definidas as métricas a utilizar na quantificação do ganho para o negócio. Nesta fase, a especificação deve ser tão completa no levantamento de requisitos quanto possível, no entanto, é frequente serem adicionados novos requisitos nas fases de desenvolvimento, testes e suporte (após entrada em produção).
3. Desenvolvimento/Configuração – Fase na qual é preparado o ambiente e configurada a ferramenta de automação (Blue Prism) segundo o desenho técnico (desenvolvimento do módulo automatizado). Esta compreende tanto a configuração dos processos como dos objetos necessários à execução das atividades definidas na fase de design. Paralelamente à configuração, são realizados testes às componentes desenvolvidas (testes unitários) e testes de integração. O desenvolvimento deverá ser realizado em módulos com a maior flexibilidade possível por forma a permitir a não só sua reutilização, mas também suportar possíveis mudanças de requisitos simples.
4. Teste & Entrega – Nesta fase, a solução desenvolvida passa por uma fase de testes de aceitação (UATs) realizados com acompanhamento do cliente e é realizada uma gestão de defeitos, após a qual é elaborado um plano de entrada em produção que é também preparada nesta fase. Caso a solução não cumpra os requisitos necessários, poderá voltar à fase de desenvolvimento por forma a corrigir ou incluir novos requisitos. Se existirem novos requisitos nesta fase, consoante o grau de complexidade e/ou a sua

importância, estes podem vir a ser apenas integrados na solução sob a forma de manutenção evolutiva (após entrada em produção).

5. Suporte – Após entrada em produção, as soluções continuam sujeitas a avaliação que permitem a realização de melhorias contínuas e a monitorização. Ocorre também a gestão de imprevistos e falhas e respetiva regularização (esta gestão é também referida como *hypercare*).

[1][9]

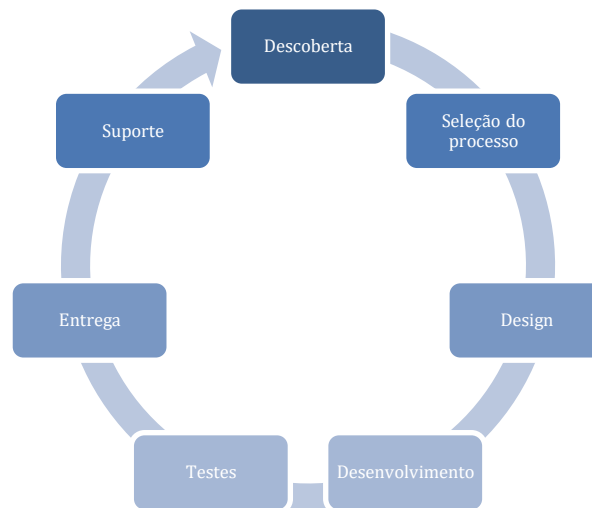


Figura 2. Gestão de Delivery

É de notar que quando existe a necessidade de acrescentar novas funcionalidades, o processo da gestão de entrega é repetido tendo as novas funcionalidades como alvo. Por exemplo, após a entrada em produção de um automatismo, é frequente existir a descoberta de novos requisitos que são priorizados e selecionados, passando por todas as fases descritas anteriormente. É também possível não serem implementados todos os requisitos numa primeira fase e ser realizada uma implementação procedimental dos mesmos, algo que está previsto ao utilizar *Agile*.

Uma das desvantagens do uso do *Agile*, no entanto, é que se torna difícil chegar a acordo com o negócio acerca das datas de entrega, uma vez que mesmo com um acréscimo de pedidos de acréscimo de requisitos, por vezes não existe a mesma flexibilidade quanto ao planeamento das entregas. Torna-se também mais difícil calendarizar uma data final e é tipicamente mais difícil manter a documentação precisa e atualizada uma vez que é frequentemente dada menor importância à mesma comparativamente às entregas de funcionalidades. Pode também ser afetada a visão do projeto como um todo devido ao foco nos sprints, um dos motivos passando pela concentração dos membros da equipa nas tarefas atribuídas para as iterações a curto prazo.

3.2 A Ferramenta

O Blue Prism consiste numa ferramenta que permite realizar automações de processos através de uma interface do tipo arrastar e soltar.

Assim, possui as vantagens das plataformas de desenvolvimento “low code”, que incluem uma menor necessidade de experiência de programação e permitem a utilização de código base, frequentemente fornecido pelas mesmas para concretizar soluções facilmente escaláveis. Estas ferramentas permitem obter mais rapidamente fluxos e funcionalidades correspondentes às necessidades de negócio pois frequentemente carecem de implementação de código de raiz. [18]

No caso do Blue Prism, são fornecidas bibliotecas de objetos e *templates* de processos que permitem desenvolver soluções sem que exista a necessidade de escrita de código.

A ferramenta apresenta funções em forma de “caixa”, geralmente apresentando inputs e outputs. Estas podem ser simples funções de inicialização de variáveis como por exemplo, um *calculation stage*, que atribui um valor a uma variável (Figura 3) ou podem conter um conjunto de eventos provenientes de um objeto, por exemplo, um *action stage* que escreve uma coleção numa folha de *Excel* num range designado (Figura 4). [14]

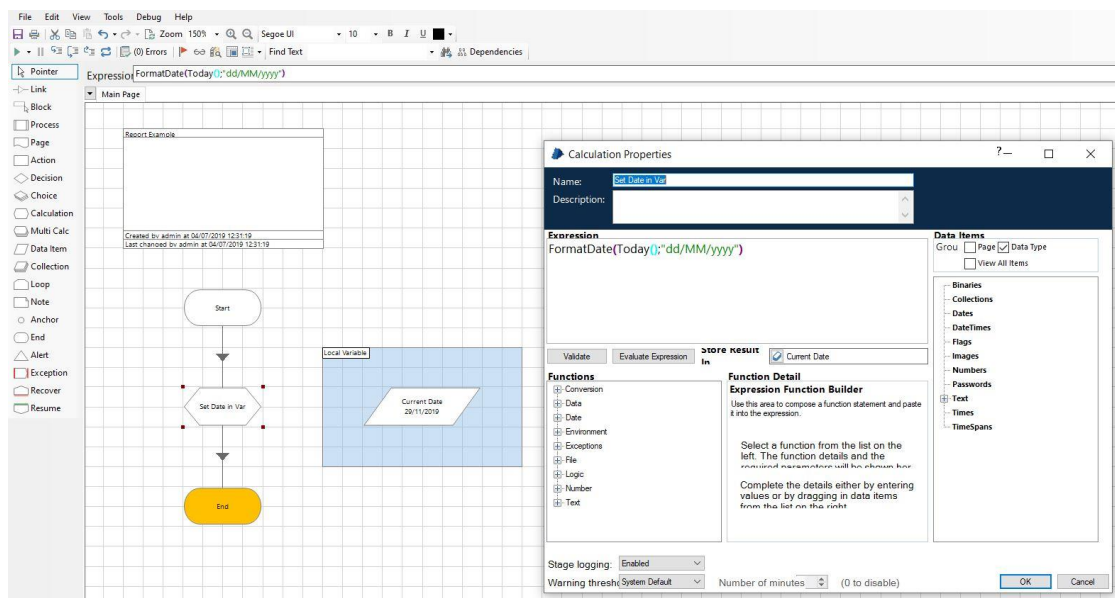


Figura 3.. Inicialização de uma variável com uso de Calculation Stage

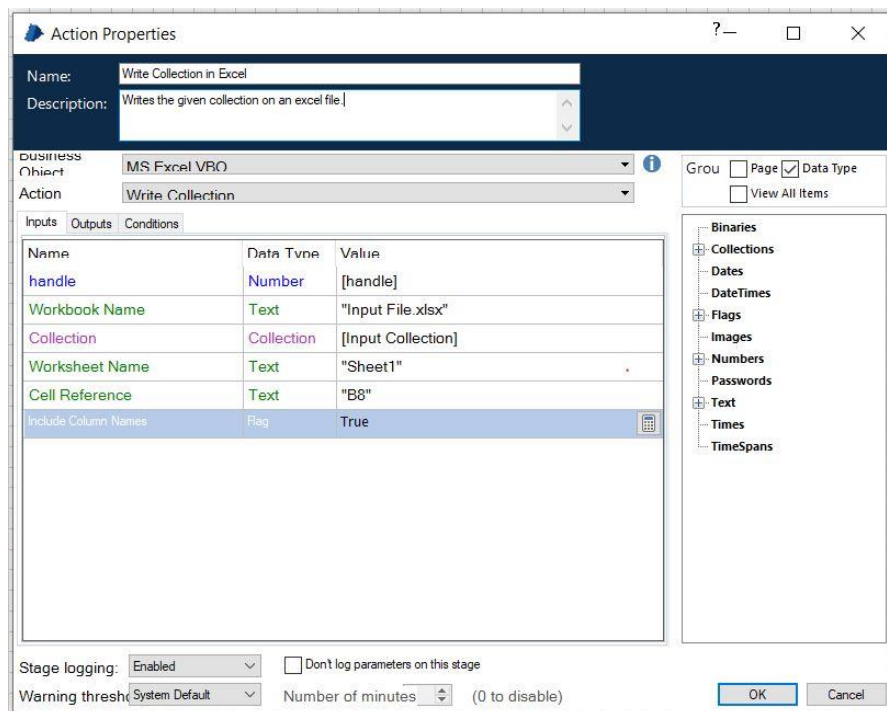


Figura 4. Action stage.

Na maioria dos processos a automatizar, não é necessário escrever código, no entanto, caso exista necessidade de utilizar uma função não existente na biblioteca nativa ou criar uma alternativa que com maior otimização, é possível o utilizador criar ações (que são representadas por *action stages*) no *Object Studio*, podendo ser não só compostas através de uma sequência de *action stages* como também de *code stages*, que permitem a escrita de funções em VB ou C# (ver Figura 3). [14]

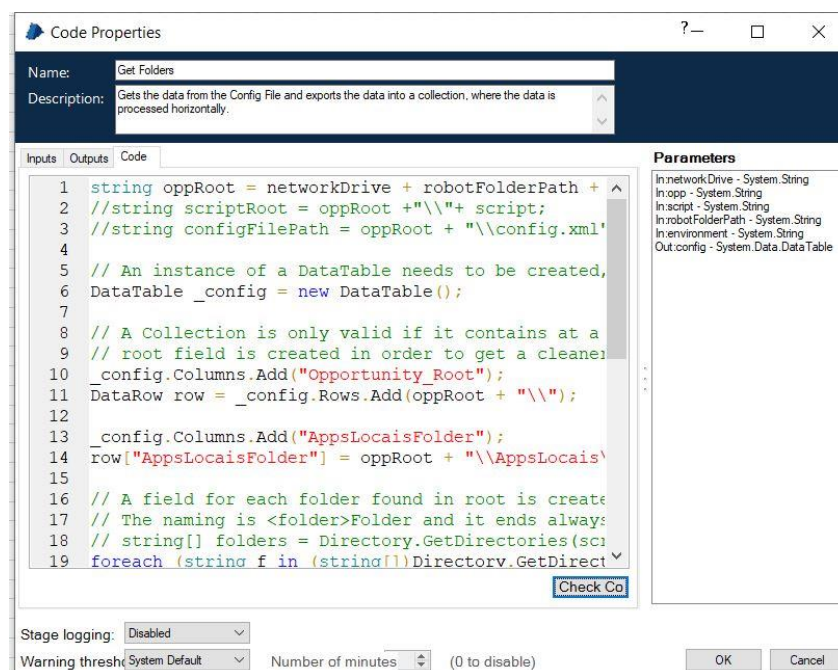


Figura 3. Code Stage

Estas ações são organizadas em objetos que podem ser reutilizados em vários processos. [13]

Quando os objetos estão associados a uma aplicação, estes obtêm uma modulação dos seus elementos e respetivos atributos, sendo que estes são utilizados para selecionar os elementos com os quais será realizada uma interação. Esta modulação pode ser realizada com vários modos, como HTML, Web Base, Accessibility, Win 32, Region Mode ou UIA. O modo de utilização mais frequente é o HTML, sendo que o Blue Prism possui ações de interação em Javascript para interação. Pode, no entanto, ser necessário escrever código de raiz caso não sejam aplicáveis.

Nos processos (*scripts*), as ações surgem como ‘caixas’ (*stages*) interligadas, formando fluxogramas que estão diretamente associados com pontos de decisão e regras de negócio. As oportunidades são estruturadas consoante os passos a realizar (à semelhança do que será descrito posteriormente por oportunidade) sob a forma de módulos os quais, dentro do *script*, correspondem a ‘páginas’ (termo do Blue Prism: *pages*).

Enquanto os objetos e respetivas ações correspondem às funcionalidades a utilizar (ex: mover ou renomear um ficheiro, fazer um cálculo de uma data ou hora, obter um valor de um texto estruturado através de expressões regulares, carregar num botão, preencher um valor num formulário), os *scripts* que os invocam possuem a lógica de negócio (ex: para que pasta o ficheiro é movido e com que nome, cálculo da data com base em X dias úteis após uma data inicial, obtenção do valor alvo, qual o botão a carregar, qual o valor a preencher e sob que condições).

A ferramenta disponibiliza também uma “sala de controlo” (*Control Room*) que permite a atribuição de tarefas a operadores digitais (referidos ao longo deste relatório como “máquinas” ou Robot) e a monitorização da execução destas em tempo real. Também permite a visualização das filas de trabalho de cada processo, permitindo avaliar para cada item destas o sucesso ou falha e exceção associada (Figura 4). [14]

Queue Contents [Clear Filter](#) Show Positions in Queue

A	Item Key	Priority	Status	Tags	Resource	Attempt	Created
✓	20190404 073041 Es	1		20190404 07304	CPX-N1B97Q	1	04/04/2019 08:30
✓	20190404 073041 Es	1		20190404 07304	CPX-N1B97Q	1	04/04/2019 08:30
✓	20190404 073041 Es	1		20190404 07304	CPX-N1B97Q	1	04/04/2019 08:30
✓	20190404 073041 Es	1		20190404 07304	CPX-N1B97Q	1	04/04/2019 08:30
✓	20190404 073041 Es	1		20190404 07304	CPX-N1B97Q	1	04/04/2019 08:30
✓	20190404 073041 Es	1		20190404 07304	CPX-N1B97Q	1	04/04/2019 08:30
✓	20190404 073041 Es	1		20190404 07304	CPX-N1B97Q	1	04/04/2019 08:30
✗	20190404 073041 Es	1		20190404 07304		1	04/04/2019 08:30
✗	20190404 073041 Es	1		20190404 07304		1	04/04/2019 08:30

Figura 4. Fila de trabalho com casos de sucesso e exceção.

Outras funcionalidades incluem um repositório de credenciais encriptadas, um repositório de variáveis de ambiente, a atribuição de *schedules*, entre outros. [14]

Por outro lado, também existem algumas desvantagens com o uso do Blue Prism. Conforme referido anteriormente, o automatismo é implementado sobre o software existente, não colmatando a dívida tecnológica caso exista. Também por esse motivo, se existir alguma alteração ao funcionamento do software, é possível que este impacte o automatismo e passa a ser necessário realizar uma ação de manutenção evolutiva (quando a alteração é planeada) ou corretiva (se a sua necessidade partir de uma alteração que já ocorreu e que tem que ser endereçada no automatismo para retomar o funcionamento correto). Esta situação torna o automatismo extremamente vulnerável a alterações e por isso é necessária coordenação com as equipas de IT do cliente.

Outra desvantagem passa pelos problemas de armazenamento de *logs*. Para cada ação realizada, é criado um registo que é armazenado na base de dados para efeitos de auditoria, no entanto, estes devem ser arquivados periodicamente por forma a evitar o risco de a encher. Infelizmente, são frequentes os problemas referentes à funcionalidade de armazenamento automático pelo que é frequente ser necessário realizar um arquivamento manual. Existe também um registo de *logs* paralelo que não possui funcionalidade aparente, mas popula a base de dados com registos desnecessários, sendo que não existe um mecanismo de limpeza para os mesmos e assim têm que ser endereçados manualmente.

3.3 Identificação de oportunidades

A fase de identificação de oportunidades (*discovery*) tem como intuito identificar casos de negócio que beneficiem de automatização. Sendo uma oportunidade, no contexto deste trabalho, um processo ao qual se procura aferir o potencial de automação.

Uma oportunidade pode ter várias motivações pelas quais se escolhe automatizar, uma das quais, a simplificação do processo. Neste caso, automatizando o fluxo mesmo parcialmente implica uma redução de complexidade que permite uma melhor compreensão deste. Esta simplificação pode também ser obtida ao analisar os passos a seguir no processo e eliminando os que não forem relevantes e que não adicionem valor ao processo. [1]

Pode também ser identificada uma oportunidade baseada em vários fluxos que podem ser padronizados, fazendo com que a sua implementação permita ao processo estar em conformidade com procedimentos ou padrões estabelecidos. [1]

Outros motivos pelos quais uma oportunidade pode ser identificada como tendo potencial de automatização, passa pela repetição dos passos a seguir no seu fluxo. A repetição

tendencialmente facilita o erro humano, sendo assim o objetivo a redução da interação humana e, consequentemente, de erros.

A alocação de tarefas à automatização permite a sua execução em menos tempo comparativamente a um humano, pelo que a identificação de oportunidades permite às equipas das áreas reorganizar os seus processos e carga de trabalho na instituição.

Alguns dos indicadores recolhidos na identificação da oportunidade incluem:

- Periodicidade da execução
- Tempo de execução
- Volumetria – nº de itens a serem processados por execução.
- FTE - *Full Time Equivalents*, 1 FTE corresponde à carga de trabalho de um recurso humano.
- Aplicações a utilizar
- Obstáculos à automatização
- Complexidade
- Custo-Benefício

Da avaliação destes indicadores pode resultar (ou não) a entrada em *backlog* da oportunidade para automatizar.

O indicador da complexidade é tido em especial atenção nesta fase do projeto dado que a proposta à entidade financeira consiste na automatização de uma oportunidade de classificação simples por sprint (ver Figura 7).

Simple	Médio	Complexo	Muito Complexo
<ul style="list-style-type: none"> • <15 Actividades, validações simples e regras • <3 Aplicações • Não Citrix ou similar 	<ul style="list-style-type: none"> • 15/25 Actividades, validações simples e regras • 3/4 Aplicações • Não Citrix ou similar 	<ul style="list-style-type: none"> • 25/35 Actividades, validações simples, regras e múltiplas excepções • 5/6 Aplicações • Citrix ou similar 	<ul style="list-style-type: none"> • >35 Actividades, validações simples, regras e múltiplas excepções • >6 Aplicações • Citrix ou similar
3 - 4 semanas de implementação	7 - 9 semanas de implementação	9 - 12 semanas de implementação	Dependente do caso

Figura 5. Avaliação da complexidade de uma oportunidade [1]. Nota: Citrix corresponde à necessidade de utilização de uma ferramenta de virtualização de aplicações ou ‘desktops’ para permitir a interação com a aplicação do cliente.

Dado que a implementação de uma oportunidade caracterizada como simples tem uma duração entre 3 e 4 semanas, é esperada a entrega de uma automatização com esta categorização por mês.

3.4 Seleção da Oportunidade

Após a fase de descoberta, na qual são avaliadas as oportunidades de automação, existe uma seleção dos processos a automatizar. Para realizar esta seleção entre as oportunidades do *backlog* disponível, é necessário priorizar. São vários os fatores que devem ser tidos em conta para a seleção:

- Impacto nos custos do serviço;
- Fatores internos/externos que impactem as prioridades (ex: disponibilidade de uma aplicação);
- Encaixe dos processos a automatizar com o calendário do processo;
- Compromissos contratuais;
- Critérios do cliente.

Após acordada a oportunidade a automatizar entre a equipa da Accenture e o cliente, é então marcada a reunião de levantamento de requisitos.

3.5 Levantamento de Requisitos

O levantamento de requisitos consiste na análise dos detalhes e requisitos de um processo por forma a identificar as suas atividades, pontos de decisão, validação, regras e exceções. É feito através de reuniões em que o processo é gravado a ser executado por um operador humano responsável pelo mesmo (*shadowing*). [1]

Para que seja possível esquematizar e realizar o desenho tanto funcional como técnico, os seguintes pontos devem ser abordados:

1. Segmentação do Processo (separação do fluxo por atividades)
2. Dados de input: dados que servem de base de trabalho.
3. Decisões: divergências de fluxo a seguir consoante a natureza dos dados ou processo.
4. Regras: exceções de negócio, casos especiais e procedimento correspondente a realizar.
5. *Triggers*: ação que despoleta o processamento do robot (ex: receção de uma solicitação, de um email, colocação de um ficheiro numa máquina específica)

6. Estruturação dos dados: forma como os dados devem ser recebidos ou entregues.
7. Resultados esperados: resultado do tratamento do item.
8. Definição de métricas a recolher.

3.6 Documentação

3.6.1 Desenho Funcional

O desenho funcional é um documento que apresenta o fluxo de atividades e o comportamento do Robot para cada situação. Deve conter ao detalhe todos os passos e informação necessários para o correto desenvolvimento do automatismo.

Este documento tem de ser aprovado pela área à qual pertence a oportunidade a automatizar para garantir que as pós-condições do funcionamento do Robot são cumpridas.

A estrutura base do documento funcional apresenta os seguintes pontos:

1. Histórico de versões – contendo a data e um resumo de cada versão do documento entregue.
2. A descrição dos ficheiros relevantes ao processo (ex: *templates*) e a identificação dos respetivos campos.
3. Fluxo de Atividades – contém um fluxograma e a descrição detalhada das atividades, regras e exceções (ver Figura 8).

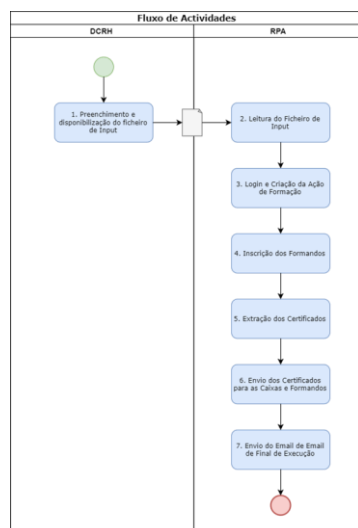


Figura 6. Exemplo de um fluxo de atividades.

4. Agendamento – indica em que dias e horas o processo deverá ser executado.
5. Métricas e respetivo exemplo ilustrativo (*dashboard*).

6. Anexos – contendo os *templates* e ficheiros de mapeamento/configuração (ver a Figura 9).



Figura 7. Exemplo de um template constante de um documento funcional.

3.6.2 Process Development Design (PDD)

O PDD é um documento alternativo ao desenho funcional. No projeto UAI, onde está a ser utilizado, difere no facto de não incluir os capítulos relativos ao agendamento (que é definido numa outra fase) e métricas (ver Figura 10), dado que parte das métricas obtidas estão definidas transversalmente e as restantes são definidas pelo cliente paralelamente à entrada em produção dos automatismos. É equivalente ao desenho funcional, não existindo muitas diferenças em relação ao conteúdo.

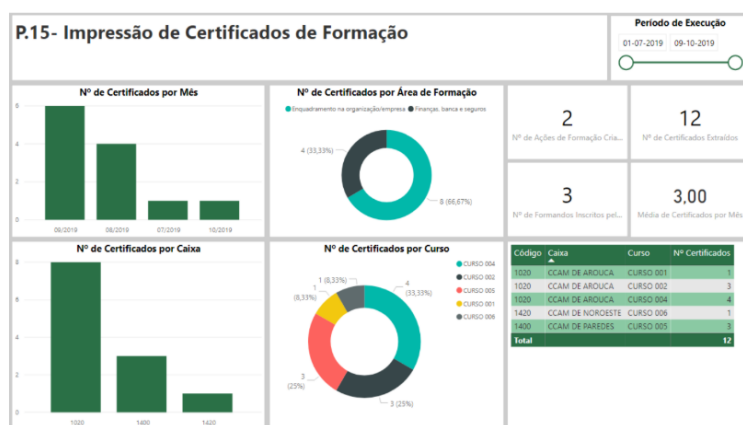


Figura 8. Exemplo de um Dashboard contendo métricas.

3.7 Desenvolvimento

Antes do desenvolvimento ser iniciado, são discutidas entre *developers* as várias abordagens possíveis à organização do *script* no contexto do Blue Prism, sendo construída a chave para as filas de trabalho e a definição dos vários passos a realizar no código.

O *developer* a cargo do desenvolvimento de uma dada oportunidade está por norma responsável pela sua estrutura, mas não prescinde de aconselhamento ou revisão por parte dos seus *peers*. É também possível a mesma oportunidade ser desenvolvida por vários *developers* sendo que nesse caso acresce a importância da atribuição de tarefas.

Após definida uma estruturação inicial, é utilizada uma aplicação de organização de tarefas (no caso do DuiT, o Trello e no caso do UAI, o Azure DevOps e AirTable) por forma a gerir tanto as tarefas como o tempo de desenvolvimento estimado e consumido nestas para cada oportunidade (ver Figura 11).

Após estarem estabelecidas na aplicação, os *developers* atribuem uma tarefa a si (ou é-lhes

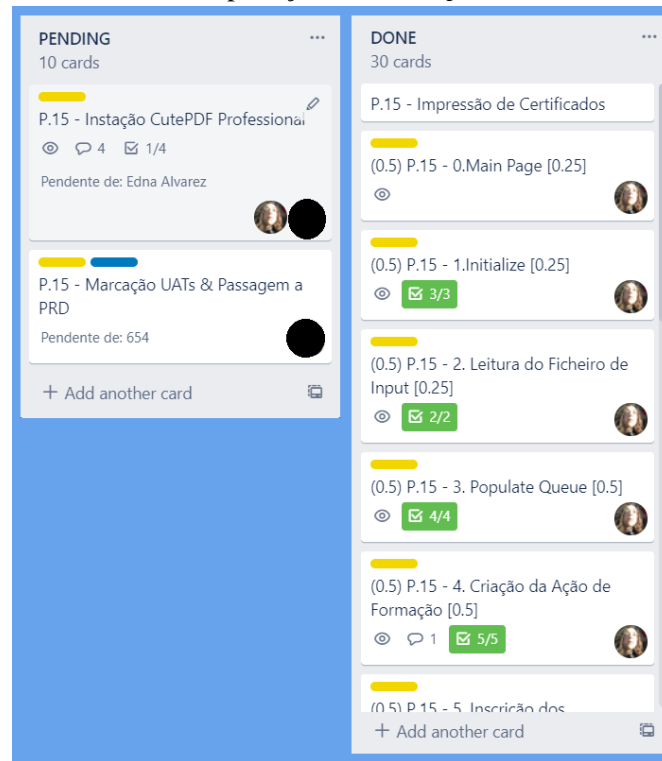


Figura 9. Exemplo da estruturação de tarefas na aplicação Trello.

atribuída a tarefa por um membro sénior ou *team lead*) para realizarem o seu desenvolvimento.

O desenvolvimento deve seguir as boas práticas. Existem boas práticas definidas pela própria Blue Prim, no entanto há que atender também às necessidades de cada projeto e processo, pelo que se podem definir novas práticas a seguir desde que estas não comprometam a segurança, complexidade, otimização ou correção do automatismo a desenvolver.

O desenvolvimento também tem a particularidade de ser tendencialmente *testdriven*, ou seja, todo o código da automatização é testado recorrentemente ao longo do desenvolvimento com o objetivo validar se cumpre os requisitos levantados no desenho funcional e que contam do caderno de testes.

3.8 Testes

Antes do início do desenvolvimento é desenhado um caderno de testes contendo casos para todos os caminhos que o automatismo deverá percorrer nas várias iterações possíveis da sua execução. O método utilizado para o desenho dos caminhos é a cobertura de caminhos primários (*Prime Path Coverage*), sendo que são também preparados todos os *datasets* necessários à realização do teste.

Durante o desenvolvimento, são realizados testes unitários às funcionalidades desenvolvidas, por norma em modo *attended*. O modo *attended* corresponde ao modo *debug*, em que é possível ter o código ‘aberto’ e ver no fluxo do processo enquanto está a ser executado, que passos estão a ser realizados. É apenas possível ter um *developer* de cada vez a utilizar este modo para o mesmo código. Neste modo, também é possível pausar a execução e fazer alterações antes de retomar a execução. Por outro lado, o modo *unattended* não permite a visualização do código, sendo os passos da execução acompanhados através de *logs* produzidos em tempo real e é possível colocar várias máquinas (ou *digital workers*) a correr o código simultaneamente.

Quando todo o fluxo do processo se encontra desenvolvido, um *tester* (que preferencialmente não tenha participado no desenvolvimento do processo por forma a evitar *bias*) realiza testes *end-to-end*. Estes são inicialmente realizados em modo *attended* passando posteriormente para *unattended* seguindo os casos explicitados no caderno de testes. Dado que é frequentemente ocorrerem mais erros de foco no segundo modo, que existem diferenças nos tempos de interação com as aplicações e que em produção as execuções ocorrem em modo *unattended*, é crucial realizar testes ao processo com recurso a este modo.

Quando a automatização passa os testes com sucesso, são então marcados os UATs (testes de aceitação) com a área cliente.

Se os testes forem realizados com sucesso, é então acordada uma data para a passagem a produção. Em caso de insucesso, são realizadas as correções e testes necessários e decorre uma nova marcação de UATs.

3.9 Métricas

Aquando da concretização do desenho funcional, são definidas as métricas do processo. Estas são fundamentais para compreender se o script cumpre os pressupostos que levaram à automatização da oportunidade selecionada e permitem não só validar os resultados como analisar a informação associada.

Embora parte das métricas sejam definidas caso a caso, existem métricas gerais que são recolhidas para cada processo:

1. Frequência do Processo
2. Tempo médio de execução
3. Tempo médio de execução por item
4. Execução mais longa
5. Nº de itens processados
6. Outros benefícios quantitativos
7. Outros benefícios qualitativos

A recolha destas métricas é feita de duas formas distintas:

1. *Query SQL* à tabela de dados interna do Blue Prism, que contém dados relativos às execuções dos scripts (ex: *resource* (operador digital/máquina) utilizada, tempo de processamento, utilizador)
2. Tabela de base de dados da oportunidade: Durante a fase de desenvolvimento, é criada uma página no processo de atualização com a função de atualizar a tabela de dados criada especificamente para cada oportunidade. A intenção desta tabela é conservar os dados e disponibilizá-los para a validação de resultados, sendo estes apresentados à entidade cliente através ou de *dashboards* criados em PowerBI ou de um relatório.

Outros indicadores passíveis de serem recolhidos (embora não sejam incluídos nos dashboards) são a quantidade de FTEs necessários à partida e a sua comparação com a quantidade de FTEs necessários após automação.

3.10 Monitorização e Estabilização

Após entrada em produção, ocorre um período de estabilização, durante o qual as execuções são acompanhadas de forma mais assertiva de modo a identificar rapidamente falhas

ou incorreções que possam pôr em causa o correto funcionamento do script. É frequente existirem casos não esperados que surjam apenas em ambiente de produção, pelo que a monitorização das primeiras execuções é essencial para detetar rapidamente possíveis falhas.

No caso de ocorrer uma falha ou for identificada uma otimização a realizar, o processo entra em manutenção, podendo ter as suas execuções interrompidas dependendo das alterações a realizar. Estas modificações são por norma realizadas em ambiente de qualidade e testadas antes de ser feita uma nova *release* para produção.

Capítulo 4 Trabalho Realizado

Devido à existência de experiência prévia com a ferramenta e o projeto, não foi necessário recorrer a um período de adaptação. Caso tal fosse necessário, a Accenture disponibiliza um período de formação no qual são disponibilizados manuais provenientes da Blue Prism assim como um ambiente de desenvolvimento no qual o formando pode adquirir os conhecimentos e experiência necessária ao desenvolvimento de soluções automatizadas. É frequente também o acompanhamento da parte de colegas mais experientes por forma a permitir uma aprendizagem

Ao longo do estágio, participei em dois projetos distintos: o DuiT, numa entidade financeira, e o UAI, numa empresa de telecomunicações.

4.1 Projeto DuiT

À data de 01/10/2019, data na qual é dado como iniciado o período de estágio no contexto da disciplina de DPEI, foi iniciado o Sprint 19.8 do projeto.

No DuiT, cada oportunidade de automatização surge com a estrutura P.X, onde X corresponde ao número de oportunidade atribuído.

Assim, a calendarização inicial é representada na Figura 12:

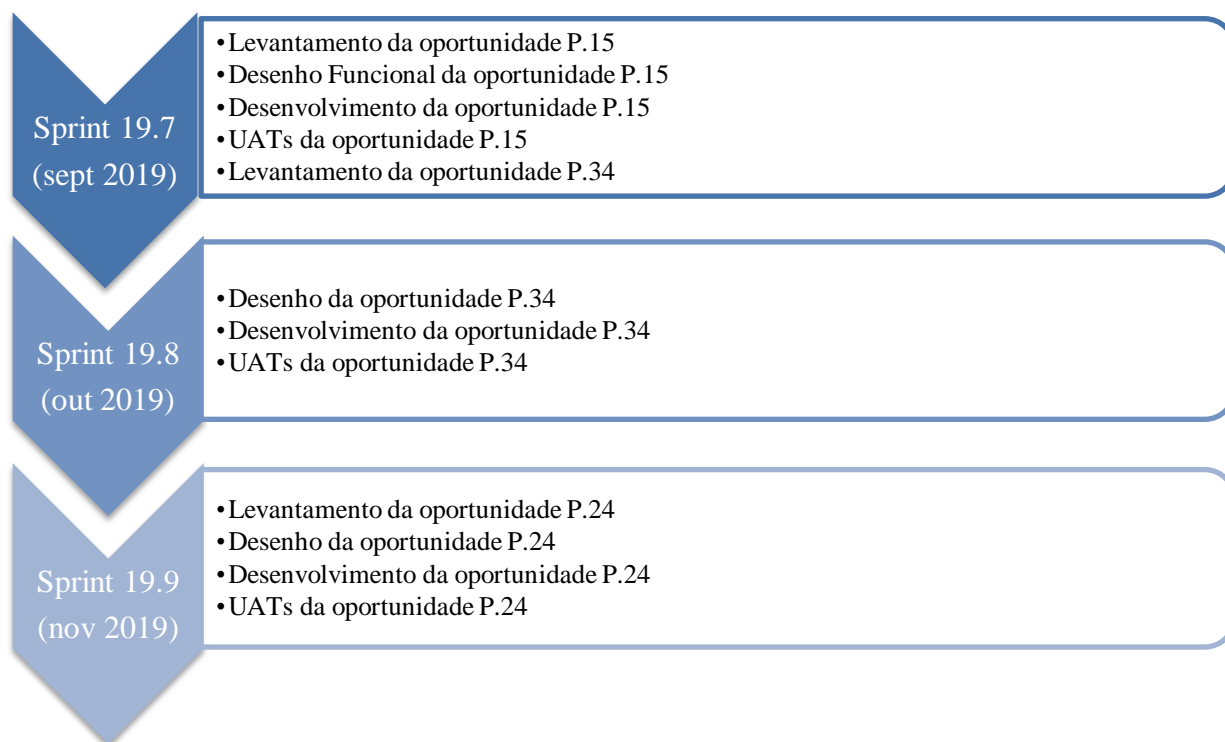


Figura 10. Planeamento inicial dos primeiros três sprints do estágio.

É de notar que apenas é apresentada a calendarização até novembro (2019) dado que esta fase do projeto estava planeada terminar em dezembro (2019). [4]

Durante o mês de dezembro (2019), era expectável realizar ações de carácter corretivo, evolutivo ou de manutenção.

4.1.1 Oportunidade P.15 – Impressão de Certificados de Formação (Sprint 19.8)

Inicialmente estava planeada a implementação do processo P.34 - Rejeitados, no entanto, devido a vários constrangimentos, a oportunidade P.15 – Impressão de Certificados de Formação sofreu atrasos. Assim, a oportunidade P.34 foi adiada para o sprint seguinte.

O desenho funcional da oportunidade P.15 foi iniciado dia 4 de setembro (2019), sendo que foi enviado um email com várias questões relativas ao comportamento do *script* no dia 11 de setembro (2019). Contudo, só foram obtidas respostas necessárias para o seu fecho a 17 de setembro (2019), o que implicou a entrega do desenho funcional mais tardia que o previsto, tendo este sido enviado dia 18 de setembro (2019).[4][5]

O documento sofreu então uma revisão tendo em conta o feedback da área, tendo só sido aprovado no dia 1 de outubro (2019).

A demora na aprovação do documento aliada à demora na disponibilização da ferramenta a utilizar para a inserção de imagens em PDF levou a um manifesto atraso na implementação deste processo (ver Figura 13).

Esta oportunidade consiste na extração de certificados de formação da plataforma SIGO (Sistema Integrado de Informação e Gestão da Oferta Educativa e Formativa) e entrega destes aos colaboradores da entidade financeira e respectivas Caixas. Está classificada como tendo complexidade “Simples” e tem uma periodicidade mensal correspondendo a 833 certificados por mês ou 40 por dia, correspondendo a 2 FTEs. Está, no entanto, estimado que no ano de 2020 processe 20.000 certificados, o dobro dos esperados no ano 2019. [6][7]

Assim, os requisitos desta oportunidade consistem em:

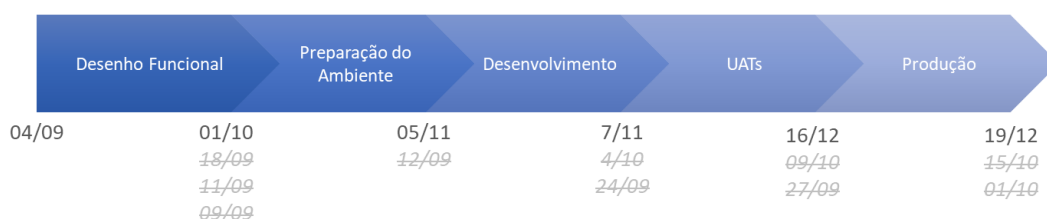


Figura 11. Milestones da oportunidade P.15.

1. Ler o ficheiro de input preenchido pela DCRH, contendo a ação de formação a criar e a lista de formandos a inscrever e aos quais extrair o certificado.
2. Verificar a existência da ação de formação fornecida no ficheiro de input, criando-a se esta não existir.
3. Inscrever todos os formandos que constem do ficheiro de input na plataforma SIGO
4. Inscrever todos os formandos inscritos no passo anterior na ação de formação recém-criada.
5. Gerar e extrair os certificados para todos os formandos e inserir o carimbo disponibilizado pela DCRH.
6. Envio do email contendo os certificados às caixas e aos formandos.
7. Envio de um email de final de execução e respetivo relatório para a DCRH.

A Figura 14 apresenta o fluxo de alto nível:

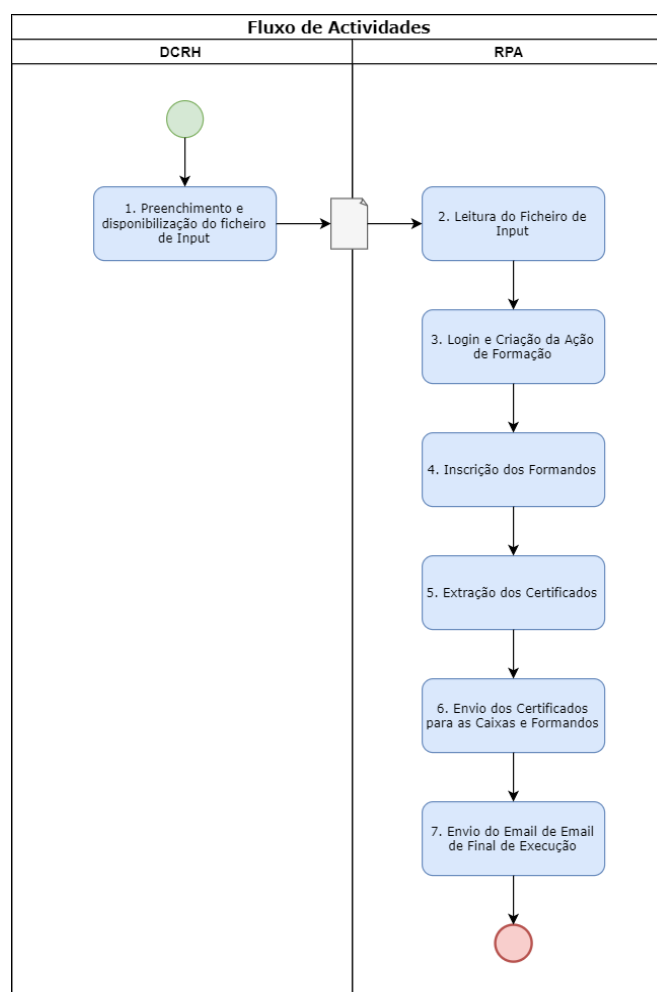


Figura 12. Fluxo de atividades da oportunidade P.15.

Esta oportunidade tinha a especificidade de não possuir uma versão de qualidade para a plataforma SIGO. Uma vez que esta plataforma não pertence ao cliente, não é possível realizar testes com inserção de dados fictícios para efeitos de teste, ao contrário de outros casos em que o cliente possui aplicações que existem somente para o efeito da realização de testes. Deste modo, uma das restrições iniciais passou pela necessidade de utilizar dados reais para o desenvolvimento das funcionalidades esperadas do automatismo.

Por forma a configurar a ferramenta Blue Prism, a implementação foi dividida da seguinte forma:

Script/Processo:

0. 'Main Page' – A página inicial contendo o fluxo *High Level*, é nesta que são colocadas as chamadas às páginas seguintes (separador do *script* contendo um grupo de passos), as variáveis de ambiente, de *startup* e globais, assim como variáveis de input locais das

páginas a chamar. É também a página que contém a recuperação de exceções mais genérica, obrigando o *script* a assumir o estado “*Terminated*” caso ocorra uma exceção não recuperável.

1. ‘Inicialização’ – Nesta página são gerados os caminhos das pastas e ficheiros a utilizar no restante processo. É também validada a existência de ficheiros de input e existência de *items* pendentes na fila de trabalho da qual depende a realização do *Launch* e *Login* na plataforma SIGO.
2. ‘Leitura de Ficheiros de Input’ – Através do uso do objeto nativo do Excel, abre o ficheiro e armazena em duas coleções distintas os dados das duas *sheets* do ficheiro de *input* (sendo a primeira referente à ação de formação e a segunda aos formandos a inscrever). Caso não exista um ficheiro de *input* na pasta alvo e a variável de *startup* “Skip Input File Reading” esteja a falso, o processo irá despoletar o envio de um email de alerta avisando para a inexistência do ficheiro e o processo termina. A existência da variável “Skip Input File Reading” permite que caso ocorra um erro de processamento durante a execução e exista a necessidade de iniciar uma nova, o email de alerta não seja despoletado dado que já não haverá necessidade de ler novamente o ficheiro.
3. ‘População da Fila de Trabalho’ – Os dados recolhidos no passo 2. Leitura de Ficheiros de Input são tratados devido ao facto de apresentarem frequentemente inconsistências ou erros que perturbam a execução se não forem endereçados numa fase inicial. As ações de formação são inseridas numa fila de trabalho própria com o estado “1 – Aguarda Criação no SIGO” com uma chave única. Os dados dos formandos são os que tendencialmente apresentam um maior número de erros ou inconsistências, sendo que algumas consistem na falta de dígitos no código postal ou ainda na formatação do seu número e/ou *check digit* de identificação. Após o seu tratamento, é assignada a cada formando uma chave única, contendo a data da execução do processo e o seu número de colaborador, ficando o item da fila de trabalho com o estado “1 – Aguarda Inscrição no SIGO”.

4. ‘Criação da Ação de Formação’ – Nesta página, o item da fila de trabalho correspondente à ação de formação é acedido. É com o uso dos seus dados que o *script* acede à plataforma SIGO e cria a ação de formação com recurso ao Objeto SIGO. Este objeto permite a interação com os diversos elementos da página. Após esta fase, o item da fila de trabalho tem o seu estado atualizado para “2 – Ação de Formação Gravada”. Existe também um passo de validação que faz parte dos requisitos apresentados pela área, sendo este a obrigatoriedade da validação do conteúdo dos campos preenchidos automaticamente pelo site nos vários separadores. Caso essa validação seja verificada, a ação de formação é submetida no SIGO e o item tem o seu estado atualizado para “3 – Ação de Formação Submetida”. A Figura 15 apresenta o *layout* da plataforma.
5. ‘Processamento dos Formandos’ – O processamento dos formandos passa por várias

Figura 13. Criação de uma ação de formação na plataforma SIGO.

fases:

- a. Inscrição dos Formandos no SIGO - Inicialmente, o desenho funcional propunha a inscrição dos formandos um a um, no entanto, ao desenvolver as ações do objeto SIGO, foi-me possível verificar que a plataforma dispunha de um mecanismo de inscrição de formandos em massa através do fornecimento de um ficheiro Excel, opção esta que permite diminuir não só o tempo de execução como o risco de exceção. Esta opção traz também a vantagem de devolver os

resultados da inscrição de cada formando (sucesso ou insucesso) e respetiva mensagem informativa de forma sistematizada. Por forma a utilizar esta funcionalidade, o *script* faz a conversão dos dados dos *items* da *queue* para os códigos esperados através do mapeamento do *template* da plataforma e submete-o, descarregando o ficheiro com os resultados das inscrições.

- b. Validação da Inscrição - Após obter o ficheiro com os resultados, o ficheiro associa cada linha de a um *item* da *queue*. Se o formando não tiver sido inscrito com sucesso, o item é marcado como exceção com a mensagem de erro disponibilizada no ficheiro de resultados. Caso o formando tenha sido inscrito com sucesso (ou já exista anteriormente na plataforma), é adicionada uma etiqueta ao formando com o número interno da plataforma SIGO associada e este passa ao estado “2 – Aguarda Inscrição na Ação de Formação”
- c. Inscrição na Ação de Formação / Ativação da Ação de Formação – O passo seguinte consiste em inscrever os formandos na ação de formação. Em termos de eficiência (navegação de menus da plataforma, número de interações com os elementos das páginas), o ideal seria sempre inscrever o formando na ação de formação e imediatamente realizar a extração do certificado, dado que esta sequência de ações tornar-se-ia mais eficiente. Contudo, não é possível extrair o certificado após inscrição se a ação de formação não se encontrar ativa, e por sua vez, esta não pode ser ativada até ter pelo menos um formando inscrito nesta. Por este motivo, após inscrição do primeiro formando na ação de formação, o *script* realiza a ativação desta, sendo o estado atualizado para “4 – Ação de Formação em Funcionamento”. Esta ativação só é realizada uma vez, pelo que após inscrição de cada formando na ação de formação, não há necessidade de repetir este procedimento. Os *items* correspondentes aos formandos inscritos passam a exibir o estado “3 – Aguarda Extração do Certificado”
- d. A Extração dos Certificados – Após cada inscrição, o *script* gera o certificado de cada formando a partir dos dados do item da *queue* correspondente, que contém a nota obtida. Posteriormente, o *script* descarrega o certificado em formato PDF da plataforma, registando o caminho do ficheiro nos dados do item. O item é atualizado para “4 – Aguarda Carimbo”
- e. Aplicação de Carimbo – Com recurso à aplicação CutePDF Pro (ver Figura 16), o *script* insere o carimbo no certificado. Para isto, é importada a imagem do carimbo que após ajustes de tamanho e posição é aplicada no ficheiro. Os *items* da *queue* passam ao estado “5 – Aguarda envio de email para as Caixas”.

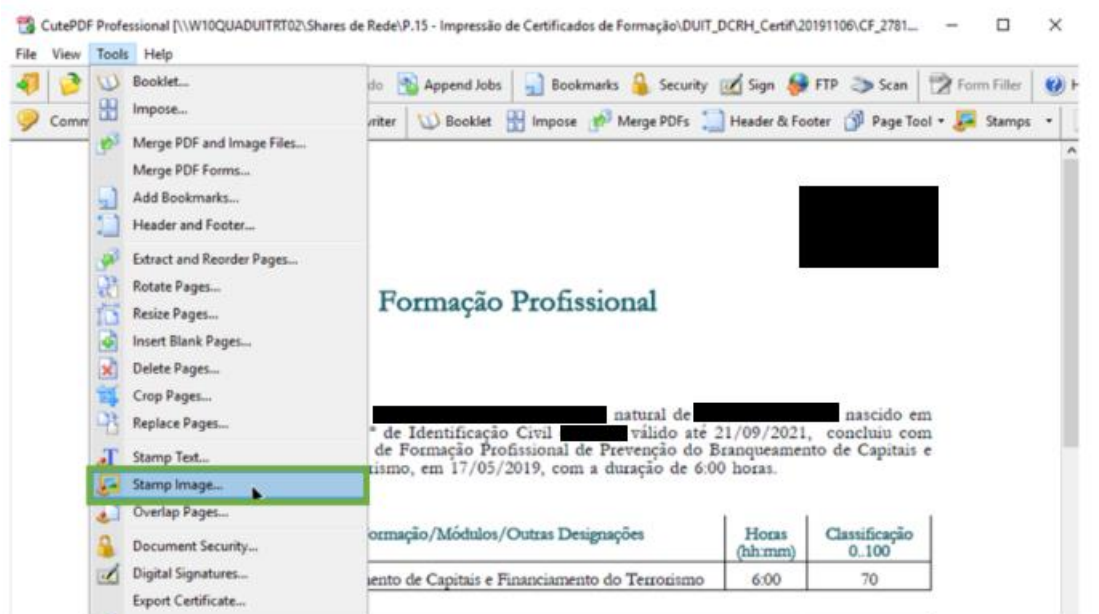


Figura 14. Aplicação CutePDF Pro para a inserção de carimbo.

6. 'Envio de Emails para as Caixas' - O *script* irá agrupar os *items* com o estado "5 – Aguarda envio de email para as Caixas" por caixa, tendo em conta os comportamentos esperados para os casos especiais. Por cada caixa, o *script* irá também enviar um email com os certificados dos formandos em anexo. O estado dos *items* será atualizado para "6 – Aguarda Envio de Email para o Formando"
7. 'Envio de Emails para os Formandos' – O *script* irá percorrer os *items* da *queue* por forma a obter o endereço de email e o caminho para o certificado de cada formando. Consoante a caixa associada, enviará o ficheiro por email ou disponibilizá-lo-á numa pasta de rede. O item tem o seu estado alterado para "7 – Certificado Enviado" e é marcado como completo.
8. 'Envio de Email de Final de Execução' – O *script* acede à *queue* respeitante aos formandos e faz uma busca por todos os *items* que tenham sido processados dentro do tempo da execução do *script*, obtendo os *Ids* internos dos *items*.

O *script* retirará informação de cada *item* acedendo a esta por *ID* e irá organizar os dados relevantes numa coleção, incluindo *items* tanto com resultado de exceção como de sucesso. Após percorrer toda a lista de *Ids* obtida, o *script* escreverá a coleção no relatório o qual será anexado ao email de final de execução. [2]

Para esta oportunidade foram criados 2 objetos de raiz, um para todas as ações correspondentes a interações no SIGO e outro para o todas as interações com a aplicação CutePDF. Foram também adicionadas ações a objetos já existentes.

Assim, para esta oportunidade foram concretizados os seguintes entregáveis:

- *Template* do Ficheiro de Input
- *Template* do Relatório de Execução
- 5 Templates com os vários tipos de emails a enviar.
- Processo P.15 (configuração da ferramenta Blue Prism)
- Objeto SIGO
- Objeto CutePDF
- Caderno de Testes
- Dashboard contendo as métricas.

O *script* foi testado ao longo de vários dias tanto durante o seu desenvolvimento (através da realização de teste unitários às ações desenvolvidas nos objetos e *script*) de forma *attended*. O desenvolvimento foi então interrompido no dia 21 de outubro (2019) dado que para este terminar apenas faltava adicionar a funcionalidade de inserir o carimbo nos certificados, função esta que estava dependente do fornecimento da aplicação escolhida pelo cliente.

Durante a interrupção do desenvolvimento, atualizei os dados do caderno de testes dado que verifiquei que os que lá estavam previamente já constavam na plataforma SIGO, e que por esse motivo não iriam permitir o teste de todas as funcionalidades do código. Por isso foi necessário aí dedicar parte das horas de desenvolvimento à manutenção do processo P.16 – Inserção de Faturas, dado que este estava a encontrar problemas quanto à configuração e acessos de utilizador.

Depois de várias tentativas de instalação e teste de várias aplicações, no dia 5 de novembro (2019) tivemos disponível a aplicação CutePDF Pro no ambiente de qualidade, pelo que foi possível terminar a implementação.

Posteriormente, foram realizados testes *end-to-end* tanto em regime *attended* como *unattended*, pelo que dei o desenvolvimento como finalizado dia 6 de novembro (2019).

Posteriormente à instalação do CutePDF Pro em ambiente de produção, os testes de aceitação (UATs) foram realizados a 16 de dezembro (2019), tendo tido sucesso e permitindo a entrada em produção no dia 19 de dezembro (2019).

Após entrada em produção, o código foi preparado para recolher os seguintes indicadores:

- N° de certificados extraídos – esta métrica irá conter o número de certificados extraídos pelo *script* do SIGO.
- N° de certificados extraídos por mês – esta métrica irá conter o número de certificados extraídos pelo *script* em cada mês.
- N° de ações de formação criadas pelo Robot – esta métrica irá conter o número de ações de formação criadas pelo *script* no SIGO.
- N° de formandos inscritos pelo Robot – esta métrica irá conter o número de formandos inscritos pelo *script* no SIGO.
- N° de certificados por Área de formação – esta métrica irá conter o número de certificados extraídos por cada área de formação.
- N° de certificados por Curso – esta métrica irá conter o número de certificados extraídos por cada curso.
- N° de certificados por Caixa – esta métrica irá conter o número de certificados extraídos por cada caixa.
- N° de certificados por Curso e por Caixa – esta métrica irá conter o número de certificados extraídos por curso e por caixa.

Sendo que esses serão disponibilizados através do *Dashboard* representado na Figura 17, criado em PowerBI.

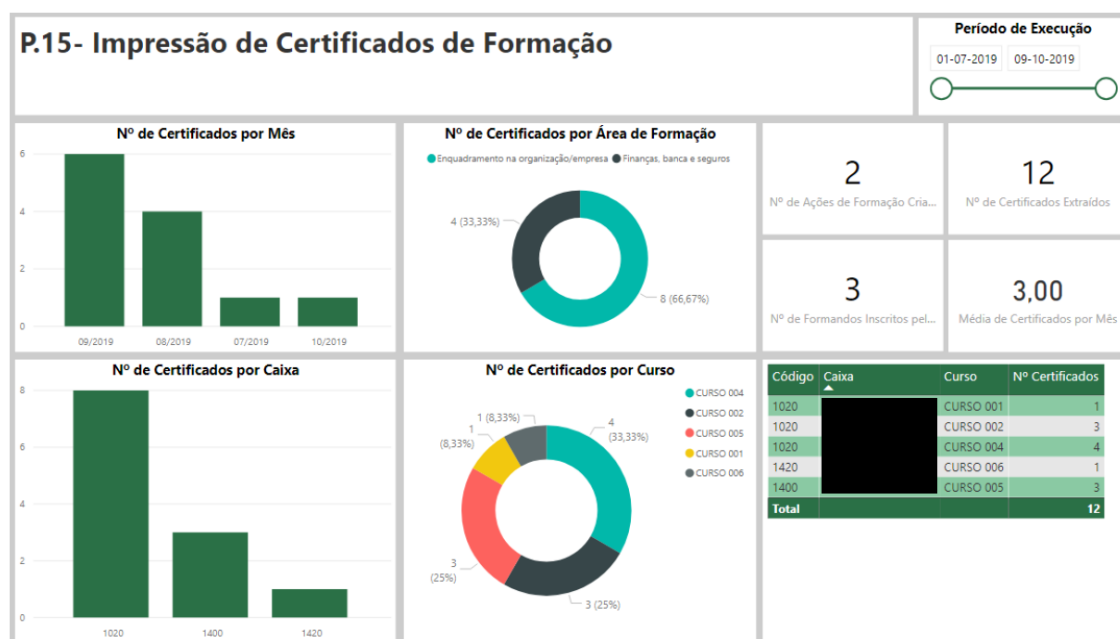


Figura 15. Dashboard contendo as métricas a oportunidade P.15 (povoado com dados de teste das execuções de qualidade).

Foi estimado inicialmente que um colaborador da área demora cerca de 10 minutos para percorrer o fluxo de trabalho necessário para criar uma ação de formação, inscrever um formando na aplicação e respetiva ação de formação e extrair o seu certificado. Após os testes realizados em qualidade, observamos que a automatização demora cerca de 3 minutos a realizar o mesmo fluxo, o que indica uma redução de 70% do tempo necessário de cada certificado a extrair.

Não foi possível recolher os resultados da implementação em produção dado ao fecho do projeto antes da primeira execução planeada, sabemos, no entanto, que a volumetria estimada para este processo em 2020 seria de 20 000 certificados ao longo do ano e que este processo possui a poupança estimada de 2 FTEs.

4.1.2 Oportunidade P.60 – Despesas Reclamadas a Bancos (Sprint 19.9)

O planeamento do sprint 19.9 passava pela implementação da oportunidade P.34 – Rejeitados.

Para esse efeito, foi realizada uma reunião a 15 de novembro (2019) no âmbito do levantamento de requisitos. Verificámos nessa reunião que o fluxo do processo não estava bem definido, algo que é fundamental para se dar início à automatização. Por outro lado, verificámos que o fluxo/processo da aplicação a automatizar ainda estava em desenvolvimento e não existia ainda uma data definitiva para a sua conclusão. Finalmente, também foi possível verificar que o âmbito da oportunidade tinha expandido quando comparado com a fase de *Discovery*, incluindo o uso de mais aplicações e um aumento das atividades, implicando um aumento de complexidade de “Simples” para “Médio”, o que implicaria que selecionando esta oportunidade, a sua implementação não estaria completa antes do final do ano.

Por este motivo, a oportunidade voltou para o *backlog* e foi selecionada uma nova no seu lugar, a oportunidade P.60 – Despesas Reclamadas a Bancos. [8]

A oportunidade P.60 – Despesas Reclamadas a Bancos consiste na consulta sistemática das despesas de todos os bancos para com a entidade financeira cliente. A partir das consultas realizadas e dependendo do tipo de tratamento esperado pelo banco, as despesas a reclamar serão enviadas ou em forma de mensagem SWIFT ou em anexo num email. As despesas são depois registadas num mapa de despesas global. Está classificada como tendo complexidade “Simples”, tem uma periodicidade mensal e está estimado que por execução consulte 429 bancos, correspondendo a 0,34 FTEs. [3]

O fluxo de alto nível da oportunidade encontra-se representado na Figura 18:

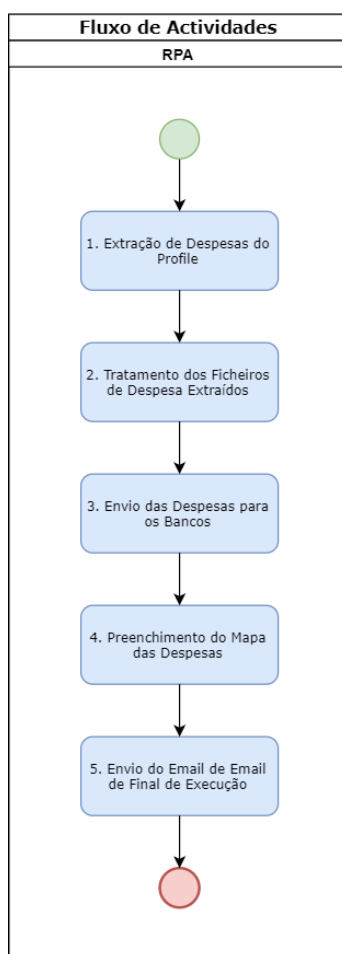


Figura 16. Fluxo de atividades da oportunidade P.60.

A reunião de levantamento de requisitos desta oportunidade deu-se no dia 20 de novembro (2019), tendo sido o documento funcional resultante entregue dia 22 de novembro (2019). Embora tenha sido dado à área um limite de uma semana para a devolução de feedback, este apenas foi recebido dia 11 de dezembro (2019) telefonicamente e posteriormente dia 12 por escrito de acordo com o nosso pedido (ver Figura 19). [7][8][9]



Figura 17. Milestones da oportunidade P.60.

Simultaneamente à entrega da documentação, fomos informados da não renovação do projeto com a equipa atual. Por esse motivo, o desenvolvimento foi iniciado sem a prévia aceitação da documentação por parte da área por forma a garantir a entrega atempada da automatização.

Desta forma, a implementação foi dividida nos seguintes passos:

Script/Processo:

0. ‘Main Page’ – A página inicial contendo o fluxo *High Level*, a main page acarreta as mesmas funções independentemente do processo. À semelhança do descrito na oportunidade P.15, é nesta que são colocadas as chamadas às páginas, as variáveis de ambiente, de *startup* e globais, assim como variáveis de input locais das páginas a chamar e a recuperação de exceções mais genérica.

1. ‘Inicialização’ – Embora partilhe as mesmas funções descritas na oportunidade P.15 no que respeita à geração dos caminhos das pastas e ficheiros a utilizar no restante processo, toma especial importância devido ao processo em si.

A consulta extensiva das despesas de bancos supera a duração de um dia útil, pelo que se torna necessário que o processo possua a capacidade de avaliar quais as extrações já realizadas e quais as extrações por realizar. Assim, a página da Inicialização verifica se a extração mensal já foi inicializada e, em caso afirmativo, onde deverá retomar.

2. ‘Extração de Despesas do Profile’ – Esta página consiste na extração e compilação da informação de dados da aplicação Profile para cada banco em lista. Inicialmente, extrai do Profile a listagem total dos bancos disponíveis para extração, guardando-a em memória. Após extração, a lista é percorrida banco a banco por forma obter todas as despesas de todos os bancos referentes aos três meses anteriores. Para cada banco, é criada uma entrada na fila de trabalho que reflete o estado de processamento e são extraídos todos os tipos de despesa para todas as moedas até 3 meses anteriores, por forma a validar se estas foram saldadas de uma forma retroativa.

Cada banco listado apresenta uma definição própria sobre o método de entrega das despesas, sendo estas: 1 – Email ou 2 – SWIFT. Estas definições são apresentadas na página de extração de despesas e são o fator de decisão relativamente ao tratamento a realizar. Ao extrair uma despesa, é despoletado automaticamente um mecanismo que envia um email com um ficheiro .xls em anexo para a caixa do robot, sendo estes armazenados numa pasta para o efeito.

3. ‘Envio das Despesas para Bancos’ – Se o tratamento tiver a definição 1 – Email, a entrega de despesas será realizada através do envio de emails os quais seguem um *template* acordado com o negócio e cujo assunto e linha introdutória são dinâmicos de acordo com a informação da despesa em anexo.
4. ‘Tratamento dos Ficheiros de Despesa Extraídos’ – Após extração, os ficheiros são renomeados de acordo com a regra definida pelo negócio: “<Banco> <Tipo de

Despesa> <Moeda>.xls” e são distribuídos num arquivo criado pelo robot para cada mês extraído. Existe a possibilidade dos bancos para os quais são enviadas as despesas requisitarem um tratamento de valores. Este implica realizar um cálculo que retire uma taxa de 4% a cada valor de custo de serviço (despesa) e separe os valores totais em valor total sem taxa, valor total da taxa e valor total. Alguns bancos também requerem a justificação dos valores dos custos dos serviços, os quais são lidos também do Profile e casos específicos, como a moeda japonesa, necessitam que as casas decimais dos cálculos gerados automaticamente sejam retiradas.

5. ‘Preenchimento do Mapa de Despesas’ – Cada despesa extraída é assinalada num mapa geral de despesas (ficheiro do tipo .xlsx) contendo todas as despesas para todos os bancos distribuídas pelos meses em que ocorreram, mês da extração, tipo de despesa e moeda. A cada mês, antes da atualização do mapa, o robot irá gerar uma cópia de segurança a qual ficará fora de acesso aos colaboradores da área por forma a evitar alterações que possam pôr em causa alteração inadvertidas dado que um dos requisitos é a possibilidade da manipulação do mapa para efeitos de cálculos.
6. ‘Envio de Email do final de execução’ – Após preenchimento do mapa de despesas, este é enviado em anexo no email de reporte de final de execução. [3]

Para este processo não existiu a necessidade de criar objetos de raiz, no entanto foram adicionadas ações ao objeto do PuTTY, correspondendo às interações com o Profile e ao objeto de Excel para permitir os ajustes e processamento do relatório dinâmico.

O desenvolvimento deste script de automatização foi caracterizado pela duração dos testes de extração, dado à ocorrência de atualizações da aplicação do Profile de qualidade que a partir de cerca das 15h não nos permitia testar os requisitos necessários. Assim, foi necessário garantir uma gestão de testes e desenvolvimento que garantissem a otimização do tempo disponível para este processo. Foi especialmente necessário garantir a finalização deste dentro do tempo de planificação com o limite máximo do dia 31 de dezembro (2019) para entrada em produção dado que após essa data o projeto acabaria.

Assim, após finalização do desenvolvimento a 17 de dezembro (2019), foram realizados os testes de aceitação a 20 de dezembro de 2019, tendo estes sido aceites embora requisitando ajustes de detalhes por parte do negócio. [8]

Finalmente, a 31 de dezembro (2019), a oportunidade P.60 entrou em produção, finalizando o projeto.

Ao preparar a entrada em produção, o código foi preparado para recolher os seguintes indicadores/métricas: [3]

- Número de extracções realizadas.
- Número de extracções por tipo de processamento – O número total de extracções efetuadas para cada tipo de processamento (1 – Email ou 2 – SWIFT).
- Número de extracções por tipo de despesa – O número total de extracções efetuadas para cada um dos tipos de despesa (NSTP ou OUR).
- Número de extracções por banco por tipo de processamento – O número total de extracções para cada banco por tipo de processamento (1 – Email ou 2 – SWIFT).
- Número de extracções por moeda – O número total de extracções por moeda.
- Valor total das taxas aplicadas por banco – O valor total de taxas aplicadas para cada banco.
- Número de tratamentos com separação da comissão – O número total de extracções com tratamento com separação da comissão.

Sendo que esses serão disponibilizados através do *Dashboard* apresentado na Figura 20, criado em PowerBI.

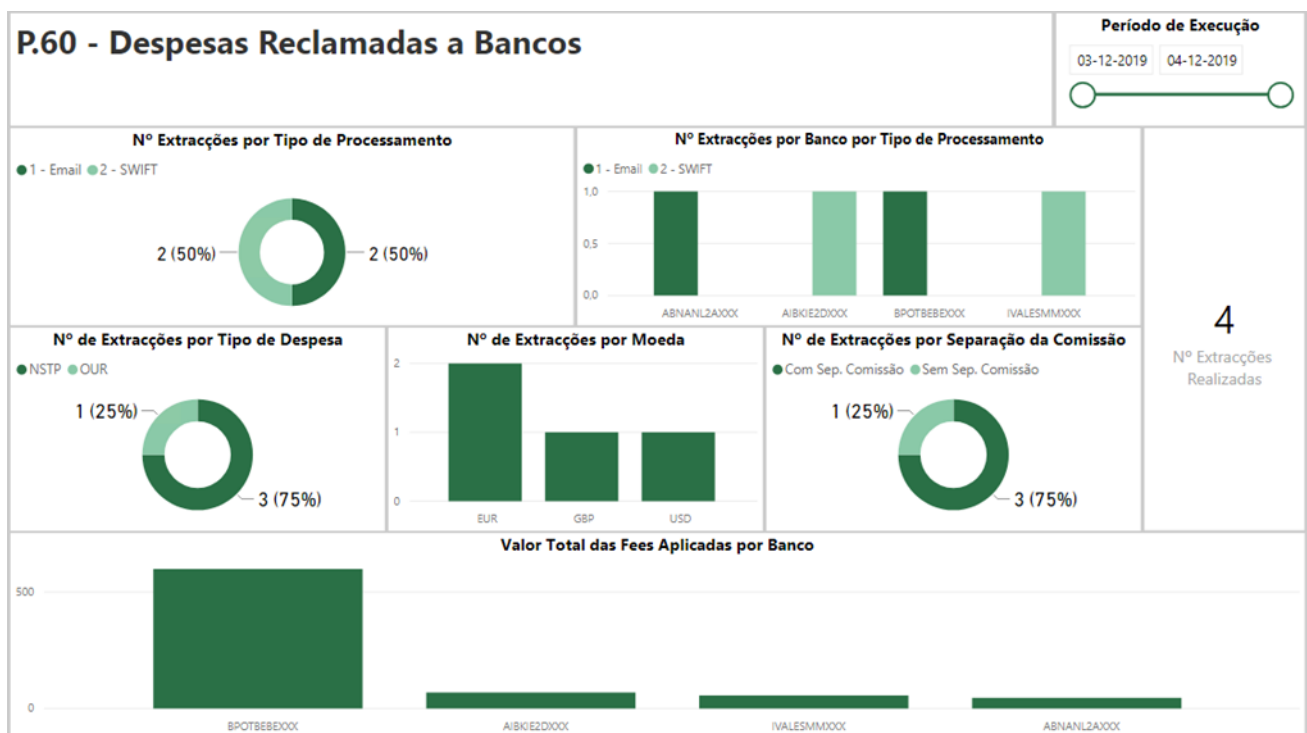


Figura 18. Dashboard contendo as métricas a oportunidade P.60 (dados teste de execuções em qualidade).

Tendo em conta que as entradas em produção de ambas as oportunidades se deram no final do mês de dezembro (2019) e o projeto terminou no final do ano de 2019, não foi possível à nossa equipa recolher indicadores, no entanto, estes estão disponíveis para apresentarem os dados pedidos à entidade financeira cliente.

Para esta oportunidade foram concretizados os seguintes entregáveis:

- *Template* do Ficheiro Mapa das Despesas
- Ficheiro Mapeamento de Configurações para edição do negócio
- 2 Templates com os vários tipos de emails a enviar.
- Processo P.60 (configuração da ferramenta Blue Prism)
- Caderno de Testes
- Dashboard contendo as métricas.

Nos últimos dias do projeto, a 30 e 31 de dezembro (2019), para além dos ajustes finais a realizar na oportunidade, foi também necessário realizar a passagem de conhecimento à equipa que viria a ficar responsável pela monitorização dos processos. Finalmente, realizei o fecho do projeto a 31, o qual incluiu a entrega da documentação final ao cliente e preparando a calendarização das execuções dos processos.

4.2 Projeto UAI

Com o término do projeto DuiT na entidade financeira, fui integrada num novo projeto a 3 de janeiro, o UAI.

Embora o âmbito do UAI seja ligeiramente diferente devido ao facto de se realizar numa empresa de comunicações, a metodologia utilizada será a mesma, apenas diferindo no tipo de documentação usada.

Assim, ao entrar no novo projeto, e dado ao facto de se esperar a chegada de novos membros à equipa, a primeira entrega que realizei foi um documento detalhando todos os passos necessários para aceder à VPN da empresa cliente e ao ambiente de desenvolvimento, tendo sido entregue no mesmo dia (3 de janeiro).

Nesta fase inicial, também me foi disponibilizado acesso ao repositório de informação dos processos onde pude verificar que o *setup* inicial do projeto se realizou no final de 2019 e qual era o planeamento para os processos a implementar.

4.2.1 Oportunidade 0002 – LAC BO

Assim, a oportunidade que me foi atribuída foi a 0002 – LAC BO. Esta oportunidade consiste na realização das portabilidades de números de outras operadoras para a operadora cliente. Para isto, o processo consiste na leitura de uma solicitação de origem, validação de dados em duas aplicações (SFA e NSOM) e criação de solicitações de desbloqueio dos números e de suporte ao agente/validação. Esta oportunidade está classificada como de complexidade média, tendo um volume mensal de atividades de 9900, sendo o processo diário e representando um total de 20 FTEs. [12]

Assim, o processo apresenta a estrutura representada na Figura 21:

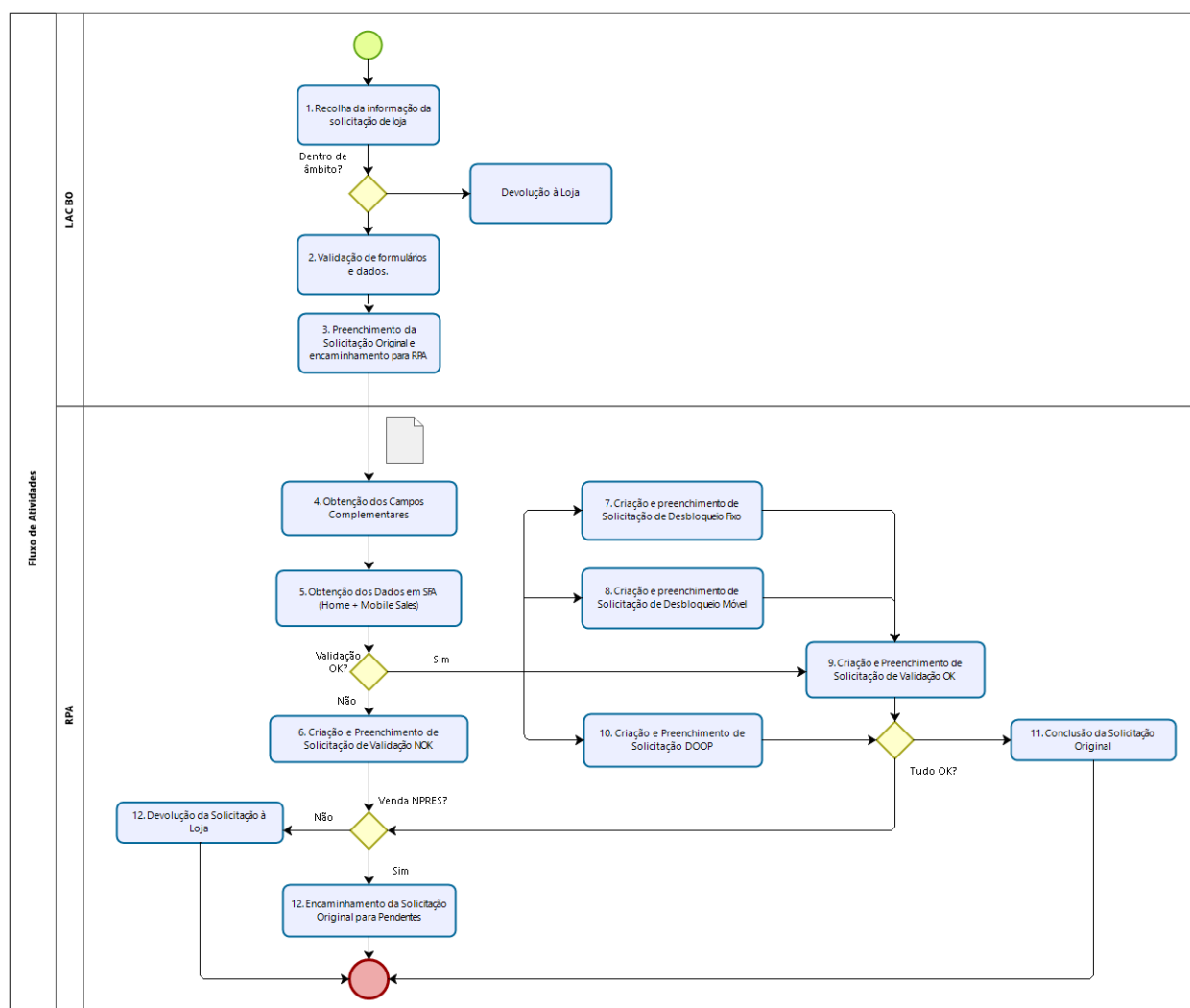


Figura 19. Fluxo de atividades

Os requisitos iniciais desta oportunidade foram levantados entre 25/10/2020 e 08/11/2020, tendo estes permitindo a elaboração de um PDD de base entregue a 15 de novembro (2019). Desta primeira versão base resultou uma iniciativa de ajuste e otimização do processo inicial da entidade de telecomunicações, o qual por um lado veio a simplificar os requisitos da automatização, mas que por outro tornou necessário realizar um novo conjunto de levantamentos ao longo do mês de janeiro por forma a obter toda a informação necessária à implementação do processo.

Esta oportunidade viu os seus objetos a serem desenvolvidos inicialmente desde novembro (2019) após entrega do PDD base por outros colegas *developers*. No entanto, os desenvolvimentos foram parados devido à falta de infraestrutura, acessos aplicativos e acesso aos ambientes necessários. Tendo sido apenas retomados em janeiro.

Após entrar no projeto, e sendo-me atribuída a oportunidade, foi necessário realizar um levantamento de condições necessárias à retoma do desenvolvimento. Esse levantamento permitiu colocar o ónus da responsabilidade no cliente, pelo que tivemos resposta e desbloqueio de vários pontos bloqueantes nos dias seguintes.

Assim, e tendo-me deparado com um *team leader* inexperiente na liderança deste tipo de projetos e mais especificamente em projetos de Blue Prism, uma das minhas primeiras interações com o projeto foi uma reunião com o manager do UAI, tendo delineado que deveria fazer um apanhado dos pontos pendentes a resolver para alinharmos ações para estabilização do projeto.

Alguns dos problemas que identifiquei foram:

- Infraestrutura inadequada à execução de processos;
- Falta de acessos nominais às VPNs para os novos elementos da equipa;
- Falta de acessos nominais com permissão de acesso remoto;
- Falta de acessos nominais e robóticos às aplicações alvo de automação;
- Falta de redundância aplicacional entre os ambientes de desenvolvimento e produção;
- Credenciais dos acessos de administração e robóticos (necessários ao desenvolvimento), *paths* de share de rede, e outras informações apenas ao alcance do team leader sob a forma de emails soltos e mensagens na aplicação Teams, levando à necessidade constante de contactar com o team lead;
- Falta da definição de uma metodologia;
- Documentação por realizar/atualizar/aprovar;

- *Developers* com pouca experiência aos quais foram entregues oportunidades caracterizadas como complexas resultando em código problemático e inconsistente;
- Prazos de entrega de automatismos pouco realistas que resultam em problemas de planeamento.
- Elevado número de exceções no código em vigor.

Os pontos referentes aos acessos e aplicações foram recolhidos num email sumário de pontos pendentes a resolver do lado do cliente. Após interação tanto com a equipa como com o cliente, percebi que um dos motivos pelos quais alguns destes pontos se apresentavam por resolver resultavam da falta de insistência nos temas. Desse modo, assumi a comunicação com dois membros do projeto do lado do cliente que permitiram após uma semana a resolução de vários dos pontos pendentes.

Estando resolvidos os acessos, tornou-se necessário gerir as credenciais destes de uma forma a permitir um uso mais facilitado. Por esse motivo, recorri de uma aplicação gestora de credencias que já tínhamos usado no projeto DuiT que permitiu armazenar e categorizar as credenciais por ambiente, aplicação e user robótico. Esta solução permitiu uma maior independência da parte dos *developers*.

Ao aproximar-se a data de entrada em produção do automatismo com identificador 0001, validei o estado da infraestrutura em produção dado que tinha encontrado questões que poderiam vir a revelar-se problemáticas. Assim, reuni-me com o *team leader* e comuniquei-lhe várias situações, incluindo a existência de *pop ups* que interfeririam com a interação com a interface, as políticas de *lock screen* e a ativação de mensagens após login, a falta de um processo *Login Agent* que permitisse o log in automático dos processos nas máquinas virtuais, a falta de atividades de arranque como o *listener* para permitir o uso das máquinas através do *Control Room*, a necessidade de configurar a pasta partilhada na rede e estrutura de subpastas utilizada em ambiente de desenvolvimento, a resolução das máquinas virtuais e os problemas de conectividade máquina-servidor.

Sendo assim, reunidas as mínimas condições necessárias à retoma dos desenvolvimentos, estruturei a oportunidade tendo-a dividido em dois processos: 0002_01 Loader e 0002_02 Executer.

Desta forma, a implementação foi dividida nos seguintes passos:

Script/Processo 0002_01_Loader:

0. ‘*Main Page*’ – A página inicial contendo o fluxo *High Level*, a main page acarreta as mesmas funções independentemente do processo. À semelhança do descrito na oportunidade P.15 e P.60, é nesta que são colocadas as chamadas às páginas, as variáveis de ambiente, de *startup* e globais, assim como variáveis de input locais das páginas a chamar e a recuperação de exceções mais genéricas.
1. ‘*Start Up*’ – Corresponde à inicialização, contém a importação do ficheiro de configuração, um ficheiro em formato .xml que contém todas as variáveis relevantes ao processo que saiam fora do âmbito de variáveis de ambiente ou globais. Esta página também é responsável por inicializar as aplicações, como o Outlook (permitindo enviar emails de alerta em caso de exceção no processo) e o CRM (a aplicação de gestão de relacionamento de clientes).
A inicialização do CRM inclui a validação da posição do utilizador (área de tratamento), dado que existem várias opções e consoante a sua escolha, as solicitações e ações permitidas que surgem são diferentes.
2. ‘*Obtenção de dados de solicitação*’ – Este passo inicia com a navegação em CRM para o menu das solicitações iniciais para a vista pretendida. Seguidamente, extrai os dados das solicitações em fila de espera, gerando um ficheiro em formato .csv cuja informação é extraída para uma coleção.
3. ‘*População da fila de trabalho*’ – Os dados da coleção obtida no passo anterior são processados, sendo retirados campos desnecessários e adicionados campos que serão necessários no *script 0002_02_Executer*. Assim, é criado um item na *queue* por solicitação a processar.
4. ‘*Close Down*’ – As aplicações CRM e Outlook são fechadas e todas as instâncias de Excel são terminadas.

Script/Processo 0002_01_Loader:

0. ‘*Main Page*’ – A página inicial contendo o fluxo *High Level*, a main page acarreta as mesmas funções independentemente do processo. À semelhança do descrito na oportunidade P.15 e P.60, é nesta que são colocadas as chamadas às páginas, as variáveis de ambiente, de *startup* e globais, assim como variáveis de input locais das páginas a chamar e a recuperação de exceções mais genéricas.
1. ‘*Start Up*’ – Corresponde à inicialização, contém a importação do ficheiro de configuração, um ficheiro em formato .xml que contém todas as variáveis relevantes ao processo que saiam fora do âmbito de variáveis de ambiente ou globais. Esta página também é responsável por inicializar a aplicação Outlook (permitindo enviar emails de

alerta em caso de exceção no processo) e obter todas as regras de validação e matrizes dos ficheiros auxiliares.

2. ‘Submain Page’ – Dada a complexidade do processo e a necessidade de controlar de uma forma mais sistemática o estado dos *items* em *queue*, estruturei a página como uma máquina de estados (ver Figura 22). Esta permite o direccionamento dos *items* consoante o seu estado o que se torna ideal no caso da interrupção ou retoma de uma execução e para o acompanhamento de *items* que não passem por todos os estados.

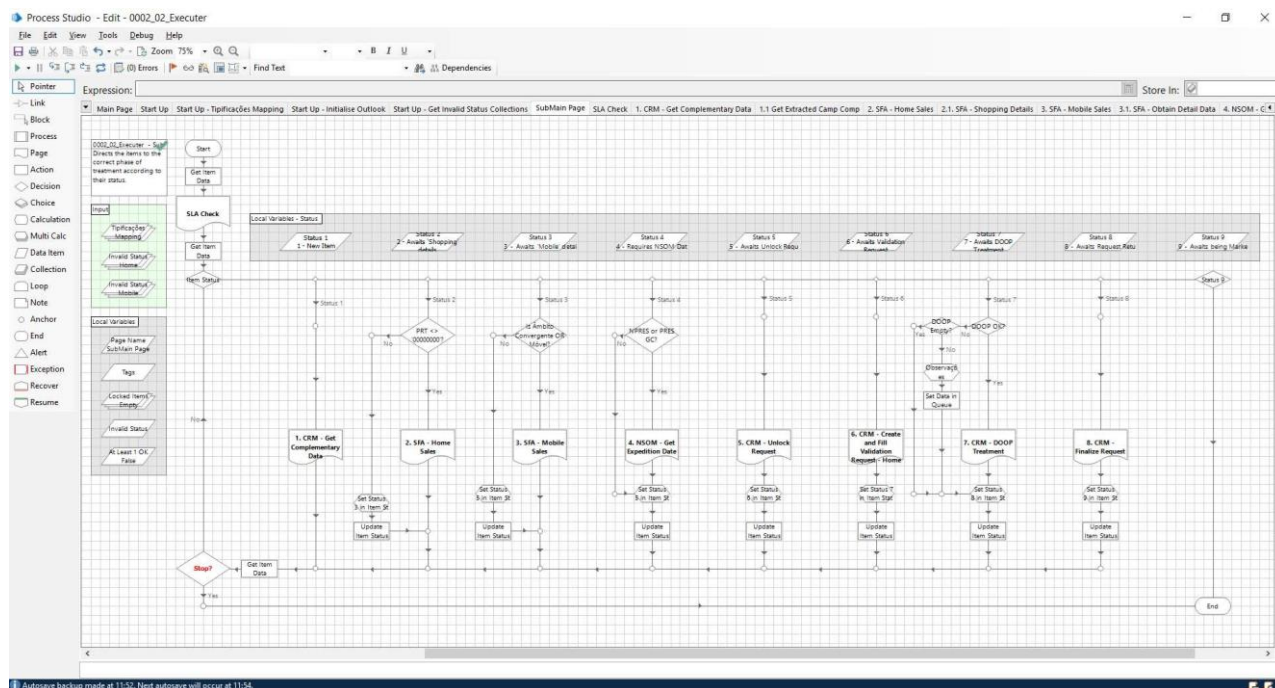


Figura 20. Página Submain estruturada com máquina de estados.

Na Submain Page, encontram-se as seguintes páginas:

- a. ‘SLA Check’ – Valida se o SLA (*Service Level Agreement*, neste caso, refere-se ao intervalo de tempo definido ao fim do qual o processamento deve passar para o cliente) máximo para processamento do item por parte do automatismo já foi ultrapassado, ou seja, se o tempo alocado para o processamento do item foi excedido. Esta funcionalidade pode ser ativada, ajustada ou desativada através do ficheiro de configuração e garante em caso de ativação, que se um item ultrapassar o limite de tempo para processamento, este seguirá o fluxo diretamente para o estado 8 e correspondentemente para a página 8. CRM – Finalização da Solicitação, sendo encaminhada para a fila de trabalho “RPA Pendentes”, onde o cliente poderá trabalhar a solicitação.

- b. '1. CRM' – Obtenção dos campos complementares – Correspondente ao ao estado 1 – *New Item*, esta fase consiste na obtenção de informação adicional da solicitação, incluindo campos como n°s de telemóvel a portar, n° de acesso fixo, resultados da validação documental para cada n° a portar, entre outros. Esta inicia com a validação da posição do user e da ordenação dos campos complementares por forma a garantir a correta extração da informação. A partir desta fase torna-se possível classificar o âmbito de um *item* como “Móvel”, se apenas apresentar números móveis, “Fixo”, se apresentar apenas um número fixo, ou “Convergente”, caso apresente ambas as modalidades. Finalmente, são realizadas validações a parte dos campos que poderão alterar o tratamento esperado da solicitação inicial (ex: falta do preenchimento de um campo obrigatório que implique a sua devolução direta no passo 8.).
- c. '2. SFA – Home Sales' – Corresponde ao estado 2 – *Awaits 'Shopping' details*, abrindo o portal de vendas na componente fixa da venda e extraíndo informação como estados da venda (que poderão ser válidos ou inválidos) e o regime da venda (Presencial ou Não Presencial). Caso o item seja de âmbito “Convergente” ou “Móvel”, este seguirá para o estado 3, caso contrário, avançará diretamente para o estado 5.
- d. '3. SFA – Mobile Sales' – Corresponde ao estado 3 – *Awaits Mobile details*, seguindo do passo anterior para validar o estado da venda móvel dos números a portar e o CVP (Código de Validação de Portabilidade) de cada um. É também verificado se cada número retirado dos campos complementares (estado 1) encontra uma correspondência no portal. Caso todos os números da solicitação original encontrem uma das seguintes condições: estado de venda inválido, CVP vazio, ou número não existente na venda, o automatismo passa diretamente para o estado 6.
- e. '4. NSOM – Obtenção da Data de Expedição' – Corresponde ao estado 4 – *Requires NSOM Data*. Caso o item tenha sido classificado como pertencendo ao regime Não Presencial, será necessário entrar no portal NSOM por forma a obter a data de fim de expedição dos números móveis a portar. Se o regime for do tipo Presencial, o automatismo não executará este passo.
- f. '5. CRM – Solicitação de Desbloqueio' – Corresponde ao estado 5 – *Awaits Unlock Request*. Este passo realiza cria e preenche as solicitações de desbloqueio (também chamadas de “tratamento”) dos números móveis e fixos quando estes cumprem todos os requisitos necessários para se realizar a portabilidade. É obrigatório para isso que o número a portar tenha um estado “OK” tanto a nível

dos formulários validados inicialmente como das validações de estado de venda e data de fim de expedição. Os números que resultam num NOK, passam diretamente ao passo 6.

- g. ‘6. CRM – Solicitação de Validação’ – Corresponde ao estado 6 – *Awaits Validation Request*. Para cada número pertencente à solicitação original, é criada uma solicitação de validação (também chamada se solicitação de “suporte ao agente”). Estas podem tomar o resultado OK ou NOK com uma breve descrição do motivo. Os motivos pelos quais os números resultam num NOK poderão ser erros de formulários, sendo que esses já irão presentes na solicitação de origem, ou erros derivados das validações do RPA, que se resumem a incorreções do estado da venda ou ausência dos números na venda correspondente.
 - h. ‘7. CRM – Solicitação DOOP’ – Corresponde ao estado 7 – *Awaits DOOP Treatment*. A solicitação original também poderá conter o resultado da denúncia ao operador (DOOP). Se esta for OK, será criada uma solicitação de denúncia. Em caso de surgir um *pop up* referindo que já existe uma solicitação de denúncia para o mesmo cliente, a solicitação mais recente será descartada e a mais antiga atualizada.
 - i. ‘8. CRM – Finalização da Solicitação Original’ – Após o tratamento dos números a portar, as solicitações podem ter um dos três seguintes tratamentos:
 - i. 1 - Concluída – Caso todos os números tenham sido portados com sucesso, a caixa de texto “Conclusão” será preenchida com os números identificadores das solicitações de desbloqueio e de validação e a solicitação será dada como “Concluída” após carregar no botão com o mesmo nome.
 - ii. 2 – Encaminhada para a Loja – Caso a solicitação seja do tipo PRES (presencial) e algum dos números resulte num NOK da portabilidade diferente do número não conter CVP, a solicitação será encaminhada de volta para a loja de origem.
 - iii. 3 – Encaminhada para Pendentes – Todas as situações nas quais pelo menos uma portabilidade seja considerada NOK e não esteja abrangida nas condições para encaminhamento para Loja serão encaminhadas para Pendentes, onde um colaborador irá realizar o seu tratamento.
3. ‘Mark Item as Complete’ – Após o fim do processamento da solicitação correspondente em CRM, o item é marcado como completo na fila de trabalho do automatismo, ficando indisponível para alterações e tendo os dados correspondentes inseridos na base de dados. Será com base nestes que se poderão retirar indicadores.

4. ‘*Mark Item as Exception*’ – Ocasionalmente, poderão vir a verificar-se erros técnicos que resultem numa exceção não recuperada (ex: timeout do tempo de resposta de um elemento da aplicação, alteração dos atributos dos elementos identificados, página indisponível, entre outros). Nestes casos, o item a ser tratado é marcado como exceção e é criada uma instância do item na fila de trabalho automaticamente para uma nova tentativa. O limite de tentativas configurável, tendo ficado definido com o limite de 3. Caso após novas tentativas o item continue com o estado de exceção, é despoletada a análise deste. Eventualmente, todos os casos de exceção serão analisados com recurso a snapshots do erro, descrição da exceção e por via dos *logs* técnicos.
5. ‘*Reset Applications*’ – Faz Logout e fecha as aplicações que estejam em aberto por forma a garantir um estado aplicacional conhecido.
6. ‘*Close Down*’ – Chama a página *Reset Applications* e prepara o ambiente para um estado conhecido para futuras execuções.

Após ter reportado os constrangimentos do projeto mencionados anteriormente, verifiquei que não tinham sido dados passos no sentido de resolver as questões referidas. Por esse motivo, parei temporariamente o desenvolvimento do automatismo 0002 – LAC BO e procurei resolver os pontos descritos. Várias situações, como algumas das políticas, instalação de *Login Agent* e criação de uma atividade de *listener* foram de resolução simples, no entanto, outras necessitaram de pedidos de intervenção da parte da equipa técnica dado que não tinha as autorizações necessárias. Outras ainda, como o problema de resposta máquina-servidor, revelou-se mais complexo, tendo sido necessário dispensar vários dias para a sua resolução.

Depois de terminar a implementação da primeira versão a ser sujeita a testes, recebemos o caderno de testes a 7 de fevereiro, tendo este sido produzido pelo cliente. Ao analisar os casos a testar, verificámos a existência de requisitos os quais não tinham sido especificados anteriormente. Surgiu também a agravante de o cliente pedir requisitos extra não acordados inicialmente cuja natureza obrigou à sua inclusão no *script*. Embora seja expectável que surjam novas especificações ao longo do processo de desenho e desenvolvimento, o número, complexidade e criticidade dos novos pedidos impactaram a solução implementada não só em termos de potencial diminuição da qualidade e consequente aumento do risco de falha, mas também em termos da complexidade estimada inicialmente, interferindo com o planeamento das actividades. Deste modo, foi necessário estender o desenvolvimento por forma a incluir os requisitos levantados na documentação e na implementação.

Tendo o código pronto, os testes de aceitação (UATs) iniciaram a 28 de fevereiro e finalizaram a 18 de março. A extensão inesperada do período de testes com o cliente surgiu de vários fatores. Entre os quais poderão ser realçados vários:

- A criação errónea por parte do cliente dos casos de teste para os UATs no ambiente de produção ao invés do de qualidade,
- A necessidade da pesquisa de dados atualizados diariamente para a criação dos mesmos,
- Pedidos de novos ajustes ao código devido a requisitos pedidos pelo cliente.
- Resolução de bugs.
- Interrupções de testes devido a indisponibilidades aplicacionais.
- Encurtamento dos períodos de teste a nível diário devido à necessidade do cliente de preparar planos de contingência devido à pandemia.
- A transição da realização de testes presenciais para a realização de testes a partir de reuniões virtuais.

Após a realização dos UATs, foi preparada a entrada em produção que se veio a realizar dia 23 de março. Tendo em conta a natureza de alguns desenvolvimentos necessários ao cumprimento dos requisitos pedidos, sugeri a sua implementação num registo de manutenção evolutiva após entrada em produção em vez de num registo de desenvolvimento.

Esta opção foi aceite, devido às vantagens que apresentava à equipa. Por um lado, este regime permitia uma entrada em produção imediata, o que permitiu mitigar mais cedo as diferenças aplicacionais entre os ambientes de qualidade e produção e evitar “mascarar” possíveis bugs derivados destas diferenças com as implementações a realizar que não seriam alvo de testes com a mesma extensão. Por outro lado, esta opção, permitindo a entrada em produção numa data inferior trazia mais benefícios do ponto de vista do modelo de negócio, influenciando os ganhos em termos económicos para o projeto.

Embora normalmente as automatizações através de Blue Prism acelerem o processamento, neste caso isso não se verifica.

O tempo médio de operação (TMO) contabilizado aquando da realização do processo por humano é de cerca de 5 minutos, no entanto, devido às verificações extra impostas pelo negócio (exemplo: validação da ordem dos campos a preencher e validação da área de tratamento do user), ao browser em uso, e à estrutura das aplicações a automatizar, o automatismo demora uma média de 15 min a processar cada item. Há que se referir também que o tempo mínimo absoluto de execução de um item foi de 4 min e 40 segundos e que o máximo foi de 47 minutos e 26 segundos, sendo o tempo máximo registado derivado da necessidade de realizar ajustes a definições do

processo na passagem a produção do código. O tempo de execução médio é também influenciado pela existência de exceções (recuperadas por mecanismos internos ou *retry* interno) e novas tentativas de processamento do item (*retry* externo).

Felizmente, a escalabilidade do processo para várias máquinas que corram em simultâneo contorna esta situação, não estando fora de questão a realização de uma otimização posterior à estabilização do processo. Outro argumento a favor desta implementação é a consistência com a qual o processamento é realizado, tendo diminuído os erros do fluxo do processo.

Por outro lado, é de referir que o processo foi desenvolvido utilizando como *browser* base o Internet Explorer dado que este apresentava uma maior estabilidade o que respeita às interações com a ferramenta Blue Prism, o que sacrifica a performance em termos de velocidade, no entanto, caso se verifique a necessidade de melhoria, não estará de parte uma manutenção evolutiva que transite os objetos para a utilização do Google Chrome, que em testes exploratórios revela maior velocidade de interação na ordem dos 70 a 80%.

Dada a conjuntura atual devido ao coronavírus, o número de *items* a processar sofreu um decréscimo massivo em relação ao previsto no caso de negócio da oportunidade. Tendo sido apenas realizados 39 processamentos entre o dia 23 de março e 23 de abril, quando seriam esperados cerca de 900, ou seja, 4,33% do esperado inicialmente. A falta de casos a processar provocou a extensão do período de *hypercare* esperado para a oportunidade (período de gestão de imprevistos e falhas e respetiva regularização).

O processo encontra-se neste momento a sair da fase de *hypercare*, na qual as execuções do robot em produção são acompanhadas com maior atenção por forma a fazer despiste de potenciais erros. Inicialmente, o processo estava a executar apenas quando existia disponibilidade da parte do cliente por forma a validar o processamento, resultando numa média de 3-4 execuções semanais. No entanto, após aprovação do negócio, o processo passou ser executado *unattended* todos os dias de forma cíclica a partir do dia 11 de maio, tendo se vindo também a verificar um aumento progressivo dos casos diários (ver Figura 23).

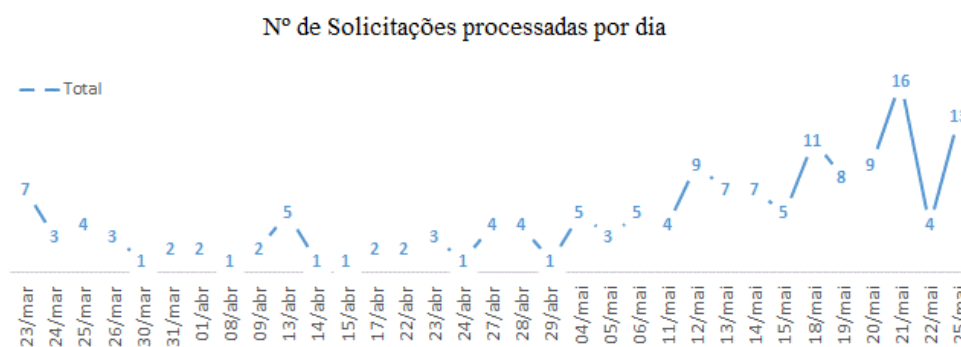


Figura 21. Nº de solicitações (casos) processados por dia

Desde a entrada em produção a 23 de março até ao dia 25 do mês de maio, foram contabilizados os seguintes valores:

- 155 *items* recebidos para processamento
- 155 *items* processados com sucesso, dos quais:
 - 4 à quarta tentativa
 - 15 à terceira tentativa
 - 16 à segunda tentativa
 - 138 à primeira tentativa

Estes corresponderam a:

- 26 casos de portabilidades de âmbito fixo
- 129 casos de portabilidades de âmbito móvel
- 17 casos de portabilidades de âmbito convergente (móveis + fixa)

Resultando em:

- 146 números a portar OK
- 44 números a portar NOK
- 9 solicitações de denúncia criadas

Para esta oportunidade foram concretizados os seguintes entregáveis:

- Legenda e capa adaptadas ao processo para o *template* do relatório de KPIs
- 2 Processos: 0002_01_Loader e 0002_02_Executer (configuração da ferramenta Blue Prism)
- 10 Suprocessos referentes às aplicações CRM, SFA, NSOM, ao browser Internet Explorer e subprocessos de reporte de execuções
- 22 objetos dos quais 10 objetos CRM, 6 objetos SFA, 5 objetos NSOM, e um IE.

4.2.2 Discovery e Shadowings

Entre os dias 16 de março e 4 de abril, participei na realização de 7 *shadowings* de oportunidades, sendo 5 destes correspondentes a novas oportunidades para avaliação da sua viabilidade, tendo sido analisados temas como acessos necessários, volumetria, passos e validações a realizar, número de aplicações a utilizar e obstáculos à automação.

Os restantes 2, foram relativos à oportunidade já em desenvolvimento por parte de um colega para uma nova recolha de requisitos dado que se verificou que a inicial não teria sido suficiente. A minha participação nestes últimos justifica-se pelo facto de estar a realizar *peer review* do código do colega referido. Embora não seja obrigatório conhecer o processo para avaliar questões estruturais e de boas práticas, a existência de conhecimento relativo ao processo a automatizar pode revelar-se útil numa ótica de correção e otimização e no caso de ter de participar no desenvolvimento.

4.2.3 Automatismo KPI_ReportBuilder

Após entrada no projeto, auxiliei na criação de um automatismo gerador de relatórios de métricas, denominado KPI_ReportBuilder com o objetivo de simplificar a recolha de dados de processamento das oportunidades automatizadas para visão própria da equipa e para partilha com o cliente. O automatismo é simples, apresentando o fluxo de alto nível apresentado na Figura 24.

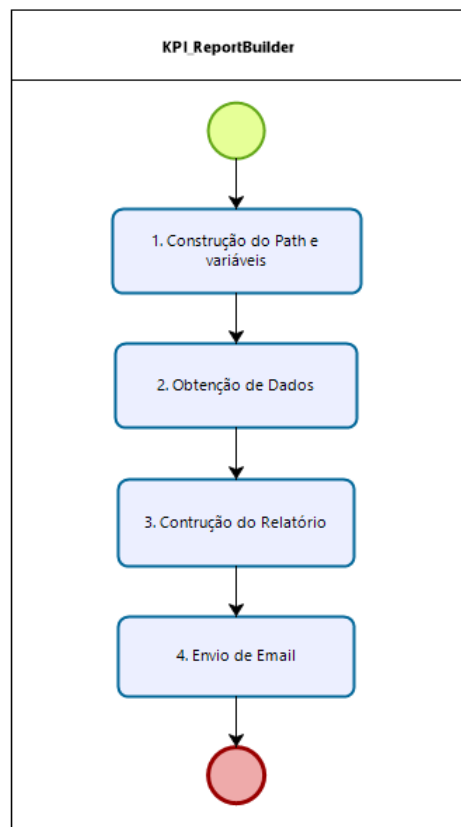


Figura 22. Fluxo de atividades do mecanismo KPI_ReportBuilder.

Por forma a garantir a sua flexibilidade para todas as oportunidades, este automatismo possui como input um identificador da oportunidade a extrair. Assim, ao fornecer o indicador, é apenas extraída a informação pertinente à oportunidade visada dado que todos os *items* processados estão inseridos na mesma tabela na base de dados. Existem também outros inputs como o *timestamp* mínimo e o *timestamp* máximo que delimitam a série de dados a extrair, estas duas variáveis são opcionais e caso não sejam definidas, o relatório extrairá todos os dados da oportunidade para o mês atual.

Assim, a estrutura do automatismo está dividida nas seguintes fases:

1. Construção do *Path* e variáveis – Este passo consiste na recolha da variável de ambiente contendo o caminho da pasta alvo para o relatório, a data e o nome do ficheiro fornecido como input para formar o caminho do ficheiro .xlsx a criar. É também aqui que são definidos os limites temporais da informação a recolher caso estes não sejam fornecidos como *input*.
2. Obtenção de dados – A partir do indicador da oportunidade fornecido como input e dos *timestamps* é gerada a *query* SQL a que irá extrair os dados da tabela alvo do SQL

Server Management Studio. Este passo inclui estabelecimento da conexão, execução da *query*, finalização da conexão e retorno dos dados como coleção.

3. Construção do Relatório – Dada a necessidade de diferenciar claramente o relatório para cada oportunidade no que diz respeito à sua capa e legenda, dividi o *template* original de registo de KPI em 3 *templates* diferentes: Capas, *Report* e Legendas. Cada *worksheet* em cada um dos três *templates* possui no seu nome o identificador da oportunidade, pelo que o automatismo cria um ficheiro em formato .xlsx e copia as folhas correspondentes à oportunidade, gerando um relatório adaptado aos *items* alvo. Após finalizada a preparação do ficheiro, os dados da coleção obtidos no passo anterior são introduzidos no ficheiro e este é guardado com o *path* gerado no passo 1.
4. Envio de Email – Este passo opcional permite, caso seja definido um endereço de receção nas variáveis de input, o envio de uma email com o *report* como anexo, evitando a necessidade de fazer login numa das máquinas que tenham acesso ao clipboard para fora da *vpn*.

Este relatório tem a particularidade de apresentar colunas com indicadores gerais, como *Start Date*, *End Date*, *Status*, *Exception Detail*, entre outros e de apresentar 10 colunas “*Custom*”, as quais podem ser preenchidas de uma forma personalizada consoante as características/necessidades da oportunidade, pelo que se torna fulcral a existência de uma folha de legenda para uma interpretação correta dos dados.

Este automatismo tem uso diário no projeto para as várias oportunidades e auxilia, entre outros, a determinar as volumetrias dos *items* para certas casuísticas e bugs.

4.2.4 Code Review

Após realização dos UATs da oportunidade 0002 LAC BO, foi-me entregue a responsabilidade de garantir o *peer review* do código dos restantes colegas como forma de garantir o *standard* de qualidade.

O processo que defini para proceder à realização da revisão do código passa os seguintes passos:

1. O *developer* cria de um ticket na aplicação de distribuição de tarefas correspondente ao código a rever (por exemplo, um processo, subprocesso ou objeto) e assigna ao *reviewer* com o estado “Review”;
2. Caso existam alterações a realizar, correções ou sugestões que possam trazer ganho, o *reviewer* detalha as mesmas no item com a justificação correspondente e assigna o

ticket de volta ao *developer*. Nesse caso, o *developer* pode realizar as modificações e submeter para nova revisão ou discutir com o *reviewer* caso tenha dúvidas ou discorde das sugestões.

3. Caso não sejam necessárias quaisquer modificações, o *developer* atribui novamente o ticket ao *developer* e altera o seu estado para “Closed”. O *reviewer* assigna ao autor do desenvolvimento para rastrear a sua autoria, o que permite à equipa identificar quem desenvolveu, o que se torna útil em caso de dúvidas na sua utilização dado ao facto de estes poderem ser partilhados pela equipa.

Algumas das validações realizadas incluem a designação de inputs e outputs corretamente nos *stages* próprios, o uso de comentários adequados no código, o uso de nomenclatura, a estruturação do código, análise de mecanismos de recuperação (qual a recuperação indicada, existência de *loops* infinitos), nomeação de *stages*, uso de variáveis locais/globais/de ambiente, entre outros.

4.3 Planeamento/Calendarização

No que respeita ao planeamento, devido a constrangimentos alheios à equipa, ocorreram alterações quanto aos processos inicialmente planeados. [7]

Estava planeado inicialmente o desenvolvimento e entrega da oportunidade P.34 durante o sprint 19.8 e da oportunidade P.24 durante o sprint 19.9 (novembro 2019). Durante o mês de dezembro (2019), correspondente ao sprint 19.10, seriam realizadas apenas atividades de caráter corretivo ou evolutivo, como se pode ver na Figura 23.

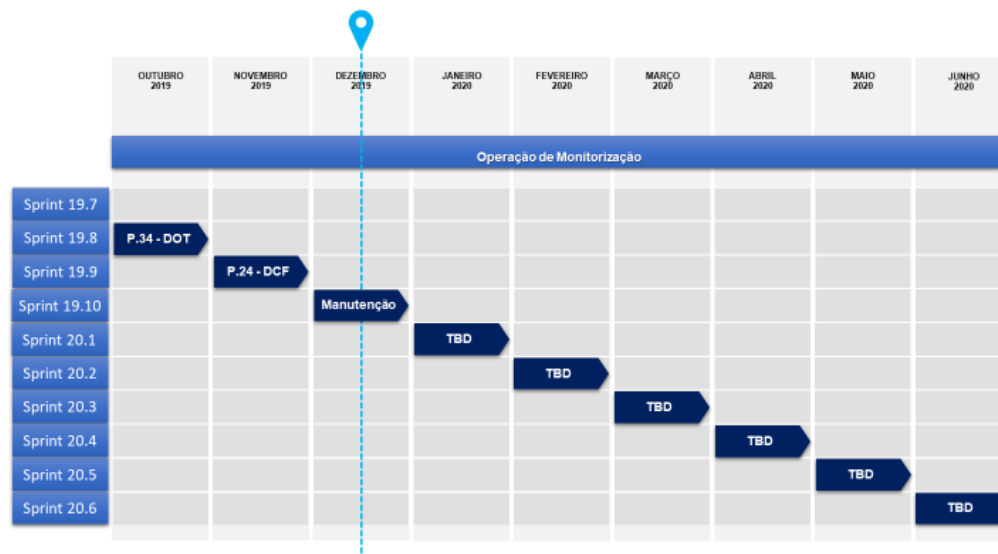


Figura 23. Planeamento Inicial

Foi verificado durante a reunião de levantamento de requisitos que a oportunidade P.34 ainda não dispunha de condições para avançar com o desenvolvimento, motivo esse que levou ao seu adiamento. Esta situação provocou no entanto um transtorno no planeamento não só por levar à alteração de prioridades no *backlog*, mas também porque, durante cerca de um mês, foram esperadas as condições necessárias para o avanço da oportunidade, pelo que ocorreu gasto de tempo que poderia ter sido utilizado em prol de outras actividades. Para evitar um alongar da situação, a oportunidade P.34 foi substituída pela P.60 dado que havia disponibilidade da área responsável para realizar a reunião de levantamento e durante a fase de discovery tinha sido identificado que a aplicação necessária já era familiar à equipa de desenvolvimento. [8]

Também por estes motivos, foi substituída a oportunidade P.24 pela P.55, dado que a priorização desta última também permitia descontinuar um dos processos *legacy* desenvolvidos por uma equipa anterior e que não possuía documentação.

Apesar de durante o mês de dezembro (2019) estarem só previstas tarefas de manutenção, devido ao atraso provocado pela reunião de condições da oportunidade P.34 foi necessário terminar o desenvolvimento da oportunidade P.60 ainda durante o sprint 19.10. [8]

No que respeita à calendarização do projeto UAI, verificou-se uma elevada discrepância entre o planeamento inicial e o desenrolar as atividades (ver Figura 26).

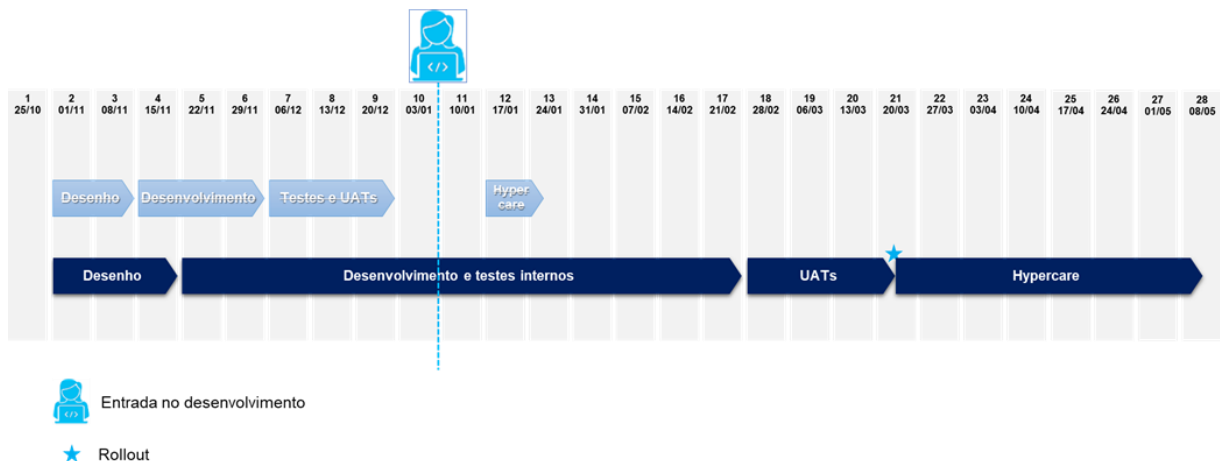


Figura 24. Mapa temporal da oportunidade 0002 - LAC BO. A azul claro está a previsão inicial e a azul escuro o tempo real consumido.

À data da minha entrada no desenvolvimento da oportunidade 0002 - LAC BO, o prazo para a sua entrega já havia sido ultrapassado.

Existiram vários motivos pelos quais essa situação ocorreu, sendo um dos quais a falta da disponibilização de acessos e infraestrutura atempadamente, tendo a situação apenas sido retificada em janeiro. A falta de infraestrutura apenas permitiu algum desenvolvimento inicial em infraestrutura própria em vez deste se realizar na infraestrutura do cliente, no entanto, não reunindo esta as condições necessárias, o desenvolvimento foi interrompido ainda em 2019.

Outro motivo do não cumprimento dos prazos iniciais foi a reestruturação do processo levada a cabo do lado do cliente, que sendo realizada em paralelo com os novos desenvolvimentos, já após a minha entrada no desenvolvimento, obrigou à realização de novas sessões de levantamentos de requisitos e ajustes ao código.

É também de salientar a ambição da estimativa inicial, que apenas contemplava 1 semana para a fase de desenho e 3 semanas para a implementação do código, esta estimativa subestimou o esforço necessário a alocar a esta oportunidade, dado que apenas teria sido adequada caso a oportunidade fosse caracterizada por uma complexidade do tipo “Simples” e não do tipo “Média”, sendo que a última necessita de um esforço estimado de 7-9 semanas de implementação (ver Figura 5). Tendo em conta que o desenvolvimento anterior não teve o aproveitamento desejado e que teve que ser em grande parte reestruturado, se se considerar o início do desenvolvimento a 3 de janeiro, data na qual a oportunidade me foi entregue, e atendendo que a finalização do desenvolvimento se deu a 28 de fevereiro, verificamos que a duração deste se encontrou dentro da estimativa de esforço estimado (8 semanas).

Capítulo 5

Discussão

O Blue Prism é uma tecnologia que permite a *developers* com conhecimentos de VB ou C# configurar soluções para automatização de processos tradicionalmente realizados por humanos. É uma ferramenta de uso intuitivo, mas que não dispensa a realização de atividades de aprendizagem por forma a dominar as várias funcionalidades necessárias ao desenvolvimento de uma solução com qualidade e a uma monitorização eficaz da execução dos processos.

Tem a vantagem de oferecer proteção de dados ao não os armazenar externamente de forma temporária, sendo que a automação manipula dados em memória. São também necessárias autorizações explícitas antes de um processo ser automatizado quando estão envolvidos dados que contenham informação pessoalmente identificável e outros dados confidenciais. Também no que respeita à gestão de credenciais de aplicativos ou contas, estas são armazenadas internamente no Blue Prism de forma encriptada e apenas utilizadores autorizados têm a possibilidade de aceder ou executar os *scripts* de automação que fazem uso destas.

Finalmente, no que respeita à integridade de dados de execução, a ferramenta também inclui o registo e monitorização de transações, onde são recolhidos dados acerca dos registo das execuções, incluindo hora de início, máquina na qual é executada, estado corrente da automação e registo de transações (*logs*) que são inicialmente armazenados na base de dados, mas que são posteriormente alvo de arquivamento por forma a impedir que a base de dados chegue à capacidade total.

No âmbito deste estágio, enquanto parte da equipa DuiT, o meu papel no POD de desenvolvimento (em que POD refere-se a uma equipa multidisciplinar) incluiu as seguintes responsabilidades:

1. Design da solução *To-Be* (Funcional e Técnico) – Desenho do caderno funcional e estruturação do processo.
2. Configuração da ferramenta e implementação (desenvolvimento do módulo automatizado), testes e respetiva entrega.

Os entregáveis consistem no desenho da solução (Desenho Funcional ou PDD), o script de automação e respetivos objetos (Processo e Objetos do Blue Prism), casos de teste e resultados (caderno de teste) e plano de implementação.

Dada a dimensão da equipa em ambos os projetos, também existe alocação dos seus elementos no POD Execução, onde as responsabilidades incluem a gestão dos processos automatizados (monitorização), suporte para a resolução de cenários de exceção, gestão de exceções, entradas em produção de *scripts* e melhoria contínua.

Os entregáveis desta fase incluem Dashboards operacionais (em PowerBI), execução e suporte das automações implementadas.

Ao final do primeiro sprint (19.7) do projeto DuiT, o desenho funcional e *templates* encontravam-se entregue e o desenho técnico estruturado. Também o caderno de testes e *dashboard* se encontrava concluído no final do sprint, houve, contudo, um atraso na entrega da configuração (*script*) provocado pela inexistência da instalação de uma das aplicações a serem utilizadas pelo *script*, algo que obrigou à paragem temporária do desenvolvimento. Após reunidas todas as condições, a implementação das funcionalidades referentes à aplicação até aí em falta e respetivos testes demoraram o total de dois dias. Este atraso não implicou alterações em demasia ao planeado. Assim, tendo esta oportunidade configurado no ambiente de produção, foram realizados os testes de aceitação com a área cliente a 16 de dezembro (2019) e a *release* do código foi migrada para produção dia 19 do mesmo mês. [8]

Dado o fecho do projeto dia 31 de dezembro (2019), e tendo em conta que estava planeado o processo executar no início de cada mês, a primeira execução corresponderia ao mês de janeiro, pelo que não obtivemos métricas em ambiente produtivo. De momento, apenas possuímos métricas retiradas em ambiente de qualidade que apontam para a redução de dez para três minutos a extração de cada certificado, o que implica um aumento significativo na rapidez do processo. [8]

Relativamente ao sprint 19.8, foi necessário realizar várias tarefas respeitantes a manutenção de processos já em produção. Também se tornou necessário auxiliar na elaboração dos *templates* da oportunidade P.55 e no desenho do seu caderno de testes. Houve neste sprint uma reunião relativa à oportunidade P.34 na qual se verificou que não teria viabilidade dentro do prazo proposto para a entrega de oportunidade por uma questão de falta de definição de fluxos e disponibilidade da aplicação a utilizar, fatores que se tornam impeditivos à automação de processos. Assim, foi realizada uma reunião referente à oportunidade P.60, dado que foi a oportunidade selecionada dado que não partilhava dos mesmos impedimentos. Da reunião,

resultou o entregável desenho funcional e respetivos *templates*, tendo esta entrega ocorrido dentro do prazo estabelecido. [8]

O maior obstáculo no desenvolvimento da oportunidade P.60 tratou-se do feedback da parte do cliente do documento entregue, que se fez tardar ultrapassando em cerca de uma semana o prazo estabelecido para alterações. Dado que na semana 16-20 de dezembro (2019) (após avaliação da necessidade de automatizar uma oportunidade) era suposto terminarmos todos os desenvolvimentos e mantermos apenas as atividades de controlo e manutenção, foi necessário realizar um maior esforço para finalizar o desenvolvimento.

Inicialmente, a minha previsão apontava dia 20 como data de finalização de desenvolvimento com apenas testes unitários, o que implicaria que a minha colega ficaria responsável pelos testes de aceitação do automatismo o qual eu tinha desenvolvido e que implicaria passagem de conhecimento. Por forma a evitar esta situação dada a lista de tarefas pendentes para fecho do ano, procurei finalizar numa data que me permitisse realizar os testes. Felizmente tive sucesso, em parte devido ao facto da oportunidade utilizar aplicações já conhecidas, pelo que dia 20 foram realizados os testes de aceitação da oportunidade P.60. A *release* para produção desta oportunidade foi realizada a 31 de dezembro (2019). Dada a realidade do final do projeto, realizei toda a configuração de *schedules* para as novas oportunidades, a passagem final de conhecimento à nova equipa que viria assumir as funções no ano de 2020 e a configuração das máquinas e acessos para entrega. [8]

Atendendo que o planeamento segue uma metodologia Agile e considerando a calendarização mais recente em vigor, os objetivos para os sprints foram cumpridos na sua larga maioria. Os atrasos verificados no projeto DuiT surgiram maioritariamente devido a fatores externos e fora do controlo da equipa.

Quando entrei no projeto UAI, a 2 de janeiro, foi-me dado acesso a toda a documentação do projeto por forma a familiarizar-me com o mesmo.

Ao realizar essa análise, deparei-me com vários problemas, entre os quais uma distinta falta de organização na documentação relativa às oportunidades, existindo várias versões dos mesmos documentos em localizações diferentes e falta de aprovações da mesma.

Este último ponto especialmente é de gravidade extrema dado que, não tendo documentação de oportunidades fechada, não conseguimos garantir que os requisitos recolhidos e a implementar ou em execução são os definidos e aprovados pelo negócio. A aprovação deste documento consiste assim numa salvaguarda de que os requisitos nele descritos foram os

acordados entre a equipa de desenvolvimento e o cliente, o que impede pedidos recorrentes e que por vezes não são compensatórios numa perspetiva benefício-custo de novos requisitos que consomem recursos e disputas nos resultados dos processamentos.

Um dos exemplos da importância do fecho deste tipo de documentação trata-se do primeiro processo desenvolvido no projeto UAI no âmbito da oportunidade com o identificador 0001. Dado que a documentação nunca foi fechada, o código sofreu de constantes pedidos de ajuste que continuaram até ao mês de maio. Embora seja expectável a existência de pedidos de alteração ou ajuste, a quantidade, complexidade e impacto destes afetaram o planeamento do projeto e a qualidade do código, tornando-o mais suscetível a falhas técnicas e incorreções funcionais. Existiram também consequências para o modelo de negócio dado que ao realizar-se um controlo de qualidade, foram disputados resultados de *items* que não tinham suporte de uma aprovação de negócio documentada. [10] [11]

Ao concluir que a falta de iniciativa por parte dos elementos responsáveis pela gestão da equipa era um dos grandes obstáculos ao sucesso deste projeto, procurei apoiar os restantes membros da equipa por forma a tentar perceber quais os temas que precisavam de maior atenção e como poderiam ser resolvidos. Aí, ficou claro que parte das questões resultavam da falta da instituição de uma metodologia, da falta de boas práticas e experiência. Procurando resolver este tema, foi realizada uma nova reunião a 20 de fevereiro tendo também participado outros colegas com experiência prévia em Blue Prism de onde resultou uma metodologia a ser implementada para uma melhor estruturação das atividades necessárias à automação das oportunidades assim como várias diretivas e práticas a serem integradas.

Finalmente, foi realizada uma iniciativa de atualização, envio e pedido de aprovação da documentação produzida.

Foram também realizadas várias sessões de acompanhamento à oportunidade 0001 – Offline com vista a reduzir o número de exceções tanto por correções ao código como por via da implementação de mecanismos de recuperação. Foi ainda criado um mecanismo de reporte de exceções que ao encontrar uma exceção tira um *screenshot* do ecrã da máquina na qual o código está a ser executado e envia-o via email como anexo para o elemento da equipa que controla as execuções dos processos (*controller*) e para os *developers*.

No âmbito da oportunidade 0002 – LAC BO, foi realizada uma reestruturação do processo no ponto de vista do negócio que viria a simplificar o fluxo de atividades automatizáveis e não automatizáveis. Apesar da diminuição de passos a automatizar, a reestruturação em paralelo

provocou dúvidas da parte da equipa do negócio que provocaram dificuldades na definição dos requisitos.

A dificuldade da sua definição veio a revelar-se consequente em duas fases. Uma durante o tempo de desenvolvimentos da oportunidade, dado que foi necessário alterar várias vezes código. E a segunda durante a realização de testes de aceitação, de onde resultaram mais requisitos não levantados em fases anteriores tanto pela rareza das ocorrências que os originaram como de pedidos do negócio fora do âmbito definido inicialmente.

Assim, foi necessário chegar a um compromisso que permitisse impedir maiores atrasos, tendo sido empregue a manutenção evolutiva para evitar maiores desvios nas datas pretendidas.

Após essa experiência, foi novamente realçada a necessidade de um levantamento correto, completo e documentado dos requisitos necessários à automação de uma oportunidade. Não só destes dependem todo o planeamento, como a diferença entre testes e execuções serem aceites ou rejeitados.

Com a situação do coronavírus, a equipa na sua totalidade trabalha remotamente recorrendo ao Teams para realizar videochamadas ou partilha de documentos. A pandemia teve efeito em termos do projeto a partir de perto do final dos testes de aceitação da oportunidade LAC BO. Com a sua entrada em produção, passei a participar nos pontos de situação com o cliente e a assumir outras tarefas dada a necessidade de voltar a priorizar as oportunidades a desenvolver tendo em conta a situação que o país enfrenta.

Esta também fez com que o *hypercare* do automatismo da oportunidade do LAC BO se tenha estendido até meio do mês de maio, sendo esta tarefa intercalada com os *shadowings* e *code reviews*. Adicionalmente, fiquei também responsável por garantir suporte à oportunidade 0001 – Offline dado que o colega responsável por esta foi atribuído a outro projeto temporariamente.

Foram vários os atrasos durante a implementação da oportunidade no projeto UAI. Estes derivaram em parte de atrasos na infraestrutura e acessos, que só ficaram resolvidos por volta de 10 de fevereiro, o que consequentemente impediu a implementação de outros automatismos e, em associação, outras entregas.

Desta forma, os entregáveis para os projetos foram os seguintes.

Para o projeto DuiT:

- 3 Ficheiros *Template*
- 1 Ficheiro de configuração
- 7 *Templates* de email
- 2 Processos

- 2 Objetos (+ ações em objetos já existentes)
- 2 Caderno de Testes
- 2 Dashboards contendo as métricas.

Para o projeto UAI:

- 1 Ficheiro de Configuração
- 1 Capa e Legenda para o template do report de KPIs
- 3 Processos: 0002_01_Loader, 0002_02_Executer e KPI_ReportBuilder (configuração da ferramenta Blue Prism)
- 10 Suprocessos referentes às aplicações CRM, SFA, NSOM, ao browser Internet Explorer e subprocessos de reporte de execuções
- 22 objetos dos quais 10 objetos CRM, 6 objetos SFA, 5 objetos NSOM, e um IE.

Considero que a integração em ambas as equipas no decorrer do estágio foi um sucesso, não só garantindo mais experiência no mundo laboral e em situações reais de negócio como tendo permitindo uma aprendizagem mútua no que respeita processos em Blue Prism. Considero que não só cumpri todas as tarefas e expectativas que me foram atribuídas como também consegui trazer outros benefícios para ambos os projetos como conhecimento prévio da ferramenta em uso e de projetos de RPA que vieram a ser úteis em fases de estabilização.

Este estágio também permitiu verificar que possíveis pontos poderão a vir causar problemas em projetos futuros.

Um dos pontos a corrigir, no caso do projeto DuiT, seria uma maior assertividade em relação às datas de resposta do cliente, dado que estes foram os principais motivos de desvio de datas-alvo. No caso do projeto UAI, seria o modelo de negócio desajustado e pouco resistente a situações extraordinárias. Para além da situação do covid que afetou o projeto, verificou-se também uma distinta falta de consideração pelos temas de manutenção evolutiva e determinação de prazos do lado do cliente, o que veio a prejudicar o ganho do projeto.

Um outro, foi a falta de liderança e experiência no *setup* e primeira fase do projeto UAI de outubro (2019) a janeiro (2020). Tendo em conta a existência de colegas com experiência neste tipo de projetos, não é direto compreender o porquê do seu suporte não ter sido requisitado quando tal é prática comum entre os projetos de RPA da Accenture. A falta da definição de uma metodologia e práticas a cumprir provocaram problemas na qualidade do código desenvolvido até aí o que levou ao consumo de bastantes recursos no sentido corretivo.

Finalmente, outro ponto a melhorar seria a necessidade da aprovação da documentação em uso no modelo operativo de uma forma mais célere, dado que impacta todo o processo de implementação e os resultados deste.

A avaliação das diversas situações descritas e a ultrapassagem destas robusteceram o conhecimento de todos os envolvidos, tendo sido fulcrais para ajustar o projeto para que este venha a ter sucesso no futuro.

Bibliografia

[1] Accenture 2018, “Definição do centro – Modelo de Governo e Operativo” – Visualizado a 17 de novembro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[2] Accenture 2019, “Desenho Funcional P.15 – Impressão de Certificados de Formação” - Visualizado a 24 de novembro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[3] Accenture 2019, “Desenho Funcional P.60 – Despesas Reclamadas a Bancos” - Visualizado a 23 de novembro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[4] Accenture 2019, “Ponto de Situação – 01 outubro 2019” - Visualizado a 17 de novembro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[5] Accenture 2019, “Ponto de Situação – 15 outubro 2019” - Visualizado a 16 de novembro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[6] Accenture 2019, “Ponto de Situação – 07 novembro 2019” - Visualizado a 16 de novembro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[7] Accenture 2019, “Ponto de Situação – 21 novembro 2019” - Visualizado a 16 de novembro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[8] Accenture 2019, “Ponto de Situação – 19 dezembro 2019” – Visualizado a 05 de janeiro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[9] Accenture 2018, “RPA Implementação e Operacionalização” – Visualizado a 17 de novembro de 2019, proveniente da solução acordada para armazenamento de documentos do projeto.

[10] Accenture 2019, “RPA PDS_DOC_20191122” – Visualizado a 3 de janeiro de 2020, proveniente da solução acordada para armazenamento de documentos do projeto.

[11] Accenture 2019, “RPA PDS_DOC_20200103” – Visualizado a 6 de janeiro de 2020, proveniente da solução acordada para armazenamento de documentos do projeto.

[12] Accenture 2020, “UAI_0002_PDD_LAC_BO_20200419” – Visualizado a 1 de maio de 2020, proveniente da solução acordada para armazenamento de documentos do projeto.

[13] Blue Prism, “Coca-Cola extends business services capacity and improves performance with RPA”, acesso a 28 de setembro de 2020, <<https://www.blueprism.com/uploads/resources/case-studies/blue-prism-cola-case-study.pdf> >

[14] Blue Prism, “Plataforma de RPA inteligente e conectada - Produto” acesso em 24 de novembro de 2019, <<https://www.blueprism.com/pt/product/intelligent-rpa-platform/>>.

[15] Blue Prism, “Process automation software drives significant improvements in key back office processes”, acesso a 28 de setembro de 2020, <<https://www.blueprism.com/uploads/resources/case-studies/shopdirect-selfservice-case-study-1.pdf> >

[16] Data Semantics, “Ui Path V/S Automation Anywhere V/S Blue Prism – Which One Is Right Platform for Your Business?” – acesso a 23 de dezembro de 2020, <<https://datasemantics.co/ui-path-v-s-automation-anywhere-v-s-blue-prism/>>

[17] Enterprise Management 360, “Top 10 Robotic Process Automation Companies” – acesso a 1 de dezembro de 2019, <https://www.em360tech.com/ai_enterprise/tech-features-featuredtech-news/top-10-rpa-companies/>

[18] G2, “Best Low-Code Development Platforms Software in 2020” – acesso a 10 de outubro de 2020, <<https://www.g2.com/categories/low-code-development-platforms>>

[19] Insider, “Examples and use cases of robotic process automation (RPA) in banking” – acesso a 1 de dezembro de 2019, <<https://www.businessinsider.com/rpa-banking-examples-use-cases>>.

[20] Jornal de Negócios (C-Studio), “Transformação Digital. Automação de Processos. O que tem a sua empresa a ganhar?” – acesso a 1 de dezembro de 2019, <<https://www.jornaldenegocios.pt/transformacao-digital/detalhe/automacao-de-processos-o-que-tem-a-sua-empresa-a-ganhar>>.

[21] Shardul Bhatt, “5 Financial Services Companies Successfully Using Robotic Process Automation” acesso a 2 de dezembro de 2019, <<https://www.botreetechnologies.com/blog/5-financial-services-companies-successfully-using-robotic-process-automation>>.

Apêndice

A. Glossário

Nome	Definição
Ação	Página de um objeto que é chamada e utilizada pelo processo.
Application Modeller	Mapeador de uma aplicação.
Attended	Modo de execução no qual o <i>script</i> é monitorizado diretamente na máquina onde está a correr. Sendo a execução realizada com observação direta do decorrer do processo.
Blue Prism (BP)	Plataforma de desenvolvimento de um <i>script</i> ou objeto
Business Exception	Erro funcional que ocorre quando uma regra de negócio não é contemplada durante o fluxo de um processo
Configuração da solução	Fase de desenvolvimento que engloba o desenho da solução, o próprio desenvolvimento da solução, seguido de testes e finalmente o período de <i>Hypercare</i>
Credencial	Contentor encriptado de dados de <i>Login</i> como <i>user</i> e <i>password</i>
Data Item	Contentor de uma variável
Descoberta de processos	Engloba a identificação e análise de oportunidades
Discovery	Processo que envolve a procura e análise de possíveis projetos automatizáveis
Ecrã	Também conhecido como <i>screen</i> , é a página atual que está a ser interagida pelo Blue Prism
Full Time Equivalent	Mede a quantidade de trabalho de um empregado
Gestão de Backlog	Engloba a consolidação de iniciativas, sua priorização e definição de um <i>Roadmap</i>
Interactive Client	Interface do Blue Prism que permite verificar informação, ver auditorias, <i>schedules</i> , etc
IT Infrastructure Manager	Garante que a infraestrutura que suporta a automação está funcional
Key Performance Indicator	Avalia o desempenho de um <i>script</i> para uma dada oportunidade
Objeto	Permite a interação entre um processo e uma aplicação
Operação de Automações	Fase de automação que inclui o <i>Scheduling</i> e monitorização de um processo, a gestão dos seus incidentes e gestão de qualquer mudança posterior ao desenvolvimento
Oportunidade	Fluxo de trabalho com viabilidade para automatização

Página	Permite agrupar vários objetos e ações dentro de um processo de uma forma lógica
Documento Funcional	Documento de descrição exaustiva do processo a ser automatizado.
Process Definition Document (PDD)	Documento de descrição exaustiva do processo a ser automatizado. Diferencia-se do Documento Funcional por apresentar o processo “as is” sem alterações ao processo provenientes da automatização.
Processo	Cria um fluxo de trabalho de uma oportunidade
Release	Pacote que contém todos os itens necessários para uma oportunidade
Resource/Máquina	Máquina virtual que executa um ou mais processos
Schedule	Permite estabelecer um horário que inicializa o processo a uma hora e dia específicos.
Script	<i>Robot</i> que executa um processo
Stage	Bloco que executa uma operação dentro de um processo ou objeto
System Exception	Erro técnico que ocorre devido a uma falha da aplicação ou do script em si
Tempo Médio de operação	Tempo médio de operação de um processo
Unattended	Modo de execução no qual o <i>script</i> é monitorizado através de uma máquina de controlo. Sendo a execução realizada sem observação direta do decorrer do processo.
User Acceptance Tests	Testes definidos pelo negócio para a solução desenvolvida de modo a garantir que cumpre todos os requisitos necessários
Variável de ambiente	Valor que afeta a execução de um processo, é guardado dentro do Blue Prism e pode ser acedido por mais do que um <i>script</i>
Work Queue/Queue	Fila de trabalho

Tabela 1. Glossário

B. Acrónimos e Siglas

Nome	Definição
BE	Business Exception
BP	Blue Prism
FTE	Full Time Equivelant
KPI	Key Performance Indicator
PDD	Process Definition Document
RPA	Robotic Process Automation
SE	System Exception
SLA	Service Level Agreement
TMO	Tempo Médio de Operação
UAT	User Acceptance Tests
VB	Visual Basic
VBO	Visual Basic Object
VPN	Virtual Private Network

Tabela 2. Acrónimos e Siglas